



## **TUGAS AKHIR - TE090362**

### **PERANCANGAN SISTEM PERGERAKAN KAMERA PADA PESAWAT *FIRST PERSON VIEW (FPV)***

Kresna Tri Jayanto Siswadi  
NRP 2211 030 064  
Mukhamad Ihya Uddin Al-Karimi  
NRP 2211 030 068

Dosen Pembimbing  
Eko Setijadi, ST., MT., Ph.D.

PROGRAM D3 TEKNIK ELEKTRO  
JURUSAN TEKNIK ELEKTRO  
Fakultas Teknologi Industri  
Institut Teknologi Sepuluh Nopember  
Surabaya 2014



***FINAL PROJECT - TE090362***

***SYSTEM DESIGN MOVEMENT CAMERA ON THE AIRCRAFT FIRST PERSON VIEW ( FPV )***

Kresna Tri Jayanto Siswadi  
NRP 2211 030 064  
Muhammad Ihya Uddin Al-Karimi  
NRP 2211 030 068

*Supervisor*  
Eko Setijadi, ST., MT., Ph.D.

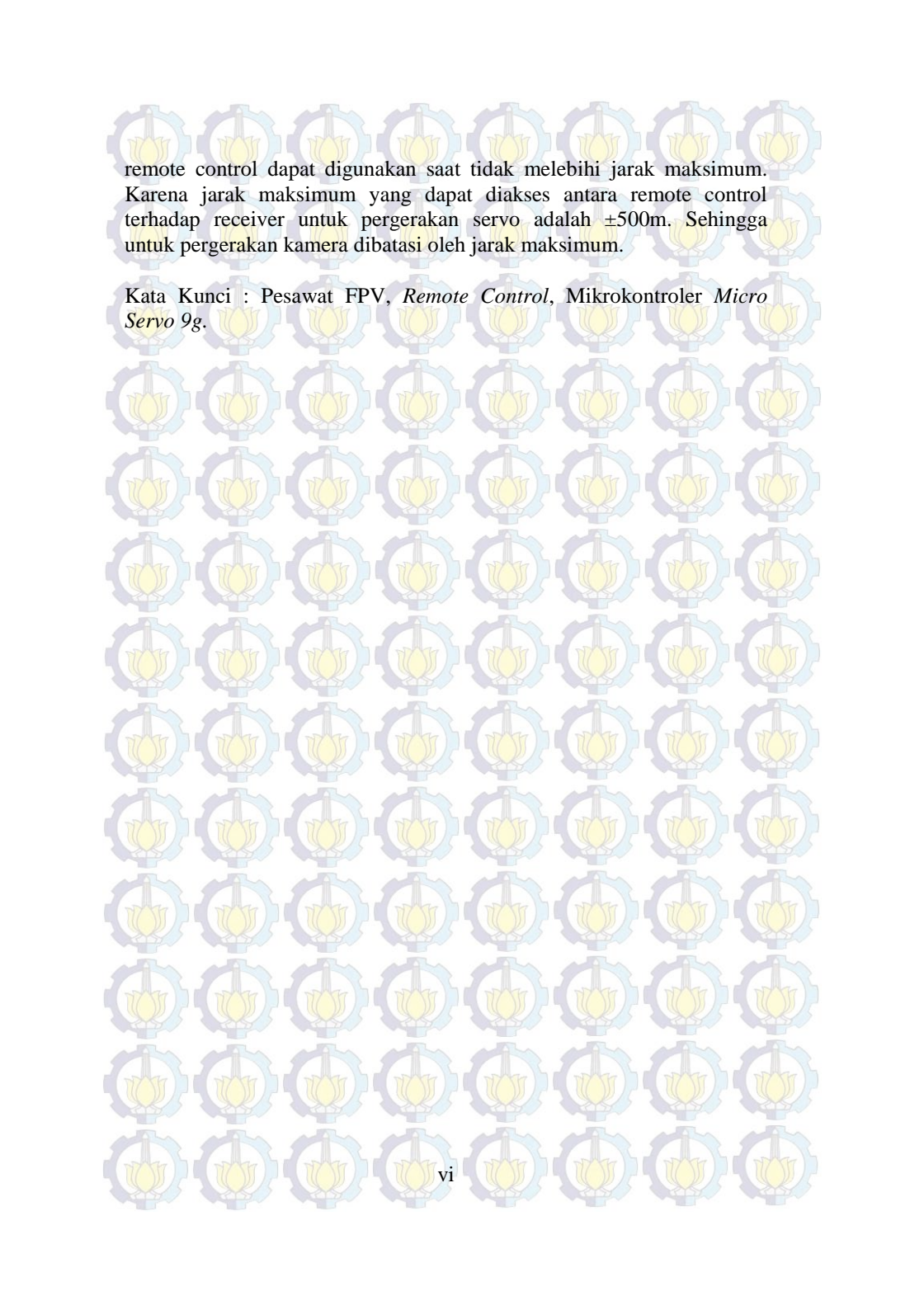
*Electrical Engineering D3 Program  
Industrial Technology Faculty  
Institut Teknologi Sepuluh Nopember  
Surabaya 2014*

## PERANCANGAN SISTEM PERGERAKAN KAMERA PADA PESAWAT *FIRST PERSON VIEW (FPV)*

Nama Mahasiswa : Kresna Tri Jayanto Siswadi  
NRP : 2210030064  
Nama Mahasiswa : Muhammad Ihya Uddin Al-Karimi  
NRP : 2210030068  
Dosen Pembimbing : Eko Setijadi, ST., MT., P.hD.  
NIP : 19721001 200312 1 002

### ABSTRAK

Pada umumnya pesawat FPV digunakan untuk kebutuhan hiburan dalam penerbangan. Sehingga pesawat tersebut kurang bermanfaat, maka dibuatlah perancangan sistem pergerakan kamera pada pesawat FPV sebagai inovasi yang bermanfaat untuk memudahkan survei udara. Untuk merealisasikan perancangan sistem pergerakan kamera ini menggunakan tombol pada *remote control* yaitu *channel 5* dan *channel 6*. Kemudian menggunakan *micro servo 9g* yang dipasang pada bagian bawah kepala pesawat dan menggunakan rangkaian penyusun yang terhubung dengan perangkat mikrokontroler. Pergerakan kamera tersebut digunakan saat pesawat diterbangkan untuk memudahkan pemotretan. Cara kerja dari sistem pergerakan kamera tersebut yaitu menggerakkan servo ke kanan 90° setiap 15° dalam 1s dengan menggunakan *channel 6* dan menggerakkan servo ke kiri 90° setiap 15° dalam 1s menggunakan *channel 5*, fungsi dari kendali remote control pergerakan servo tersebut untuk memudahkan agar pemotretan dapat dilakukan secara leluasa tanpa mengendalikan pesawat terlebih dahulu untuk arah kanan dan kiri. Kemudian untuk menggerakkan servo ke bawah 90° yaitu dengan menekan ke arah bawah pada *channel 5* dan *channel 6* secara bersamaan setiap 15° dalam 1s dan menggerakkan servo ke atas 90° dengan menekan ke arah atas pada *channel 5* dan *channel 6* secara setiap 15° dalam 1s, fungsi dari kendali remote control untuk pergerakan servo tersebut untuk memudahkan pemotretan dalam keadaan pesawat naik dan turun. Sehingga pergerakan kamera tersebut menghasilkan gerakan sesuai dengan keinginan pada *remote control*. Pengaruh dari remote control terhadap pergerakan servo tersebut yaitu



remote control dapat digunakan saat tidak melebihi jarak maksimum. Karena jarak maksimum yang dapat diakses antara remote control terhadap receiver untuk pergerakan servo adalah  $\pm 500\text{m}$ . Sehingga untuk pergerakan kamera dibatasi oleh jarak maksimum.

Kata Kunci : Pesawat FPV, *Remote Control*, Mikrokontroler *Micro Servo 9g*.



# ***SYSTEM DESIGN MOVEMENT CAMERA ON THE AIRCRAFT FIRST PERSON VIEW ( FPV )***

***1<sup>st</sup> Student Name*** : Kresna Tri Jayanto Siswadi  
***NRP*** : 2210030064  
***2<sup>nd</sup> Student Name*** : Muhammad Ihya Uddin Al-Karimi  
***NRP*** : 2210030068  
***Supervisor Name*** : Eko Setijadi, ST., MT., P.hD.  
***NIP*** : 19721001 200312 1 002

## ***ABSTRACT***

FPV aircraft generally do not have the full feature, only used for in-flight entertainment. So that the plane is less than helpful, then be made to the design of systems on the aircraft FPV camera movement as a useful innovation to facilitate aerial surveys. For realize this camera movement system design using the buttons on the remote control that is channel 5 and channel 6. Then use micro servo 9g mounted on the lower part of the head of an aircraft and use the compiler circuit connected to the microcontroller device. The camera movement is used when aircraft were flown for easy shooting. The workings of the camera movement system which moves the servo to the right 90 every 15 in 1s by using channel 6 and the servo moves to the left 90 every 15 in 1s to use channel 5, the function of the control remote control the servo movements so as to facilitate photo shoots can be done freely without controlling the plane first to the right and left. Then to move the servo down with the push towards the bottom on channel 5 and channel 6 simultaneously 90evry 15 in 1s and move the servo to over 90 by pushing towards the top on channel 5 and channel 6 simultaneously 90evry 15 in 1s, the functions of the remote control to control the movement of the servo to facilitate shooting in a State plane up and down. So that the camera movement produces movement in accordance with the wishes of the remote control. The influence from the remote control against the servo movement, namely a remote control can be used when it does not exceed the maximum distance. Because the maximum distance that can be accessed from the remote control to the

receiver to the servo movement is 500 m. So for the camera movement is limited by the maximum distance.

*Keywords: FPV Aircraft, Remote Control, Microcontroller, micro servo*  
9g.

**PERANCANGAN SISTEM PERGERAKAN KAMERA PADA  
PESAWAT *FIRST PERSON VIEW (FPV)***

**TUGAS AKHIR**

**Diajukan Guna Memenuhi Sebagian Persyaratan  
Untuk Memperoleh Gelar Ahli Madya  
Pada**

**Bidang Studi Komputer Kontrol  
Program Studi D3 Teknik Elektro  
Fakultas Teknologi Industri  
Institut Teknologi Sepuluh Nopember**

**Menyetujui :  
Dosen Pembimbing.**



**Eko Setijadi, ST., MT., Ph.D.**

**Nip. 197210012003121002**

**SURABAYA  
JULI, 2014**



## KATA PENGANTAR

Puji syukur kami panjatkan kepada Allah Subhanahu Wa Ta'ala yang telah memberikan rahmat dan hidayah-Nya. Dan tak lupa kami haturkan sanjungan kepada Rasulullah Shalallahu 'Alaihi Wassalam sehingga kami dapat menyelesaikan buku Tugas Akhir ini yang berjudul "Perancangan Sistem Pergerakan Kamera Pada Pesawat First Person View (FPV)".

Dalam penyusunan laporan Tugas Akhir ini, kami banyak mendapatkan bantuan dan dukungan dari berbagai pihak. Oleh karena itu, penulis dengan tulus ikhlas menyampaikan banyak terima kasih kepada:

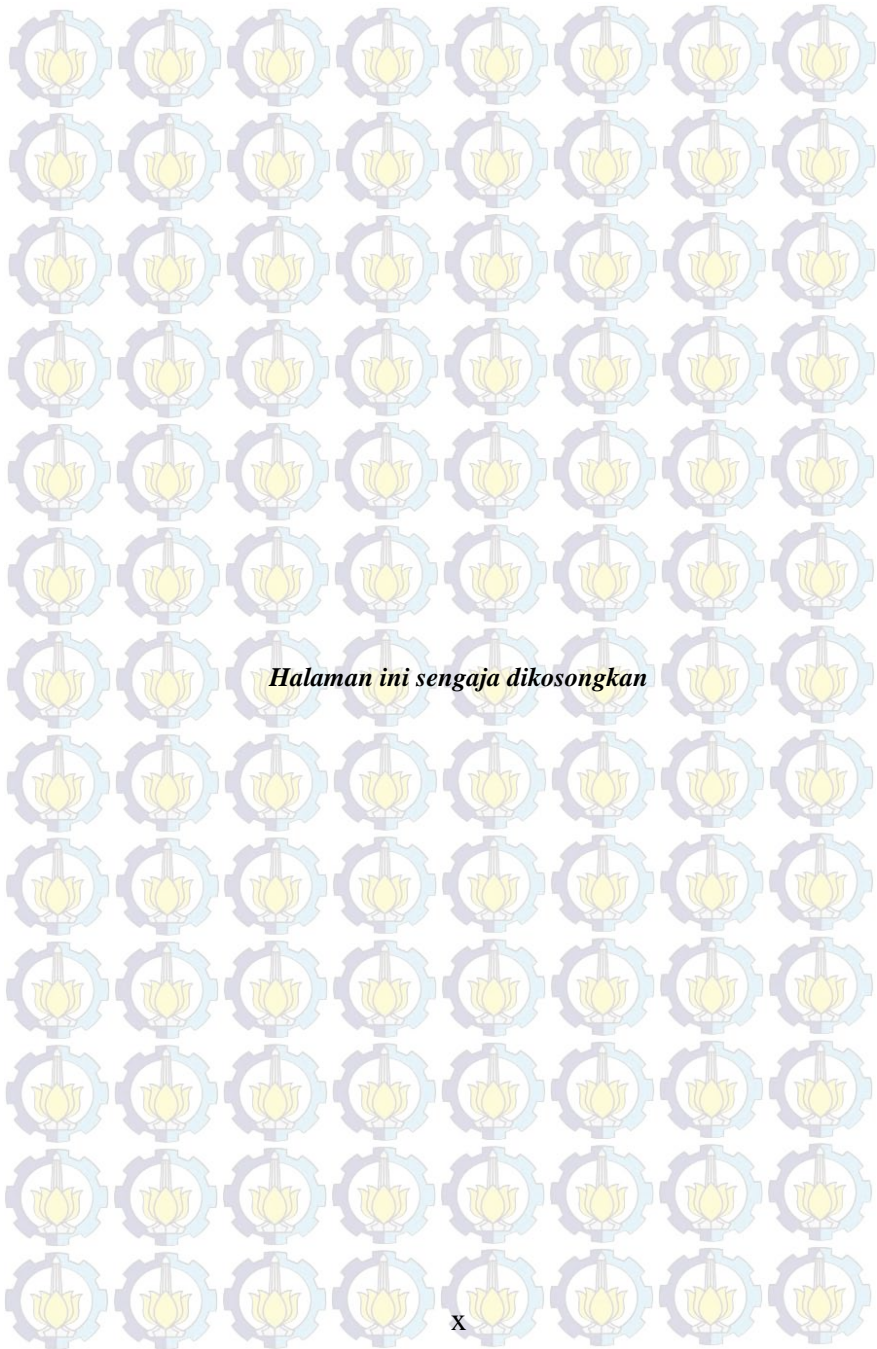
- Ayah dan Ibu kedua penulis yang telah memberikan dukungan moral, material, serta doa.
- Eko Setijadi, ST.,MT.,Ph.D selaku dosen pembimbing Tugas Akhir kami, atas segala kesabaran dan kesediaannya meluangkan waktu untuk membimbing serta memberi dukungan sehingga Tugas Akhir ini dapat terselesaikan.
- Eko Setijadi, ST.,MT.,Ph.D, selaku Kaprodi D3 Teknik Elektro, Komputer Kontrol FTI - ITS.
- Semua pihak yang tidak dapat kami sebutkan satu persatu yang telah memberi dorongan dan bantuan dalam menyelesaikan Tugas Akhir ini baik secara langsung maupun tidak langsung.

Kami menyadari bahwa pembuatan laporan serta tugas akhir kami belum sempurna, karena kesempurnaan hanyalah milik Allah Subhanahu Wa Ta'ala. Untuk itu sekiranya mohon maaf atas kekilafan kami apabila terdapat kesalahan dalam pembuatan laporan ini. Besar harapan kami untuk memaafkan kurang sempurnanya pembuatan laporan tugas akhir ini.

Surabaya, Juli 2014

Penyusun





*Halaman ini sengaja dikosongkan*

## DAFTAR ISI

<b>JUDUL</b> .....	i
<b>PENGESAHAN</b> .....	iii
<b>ABSTRAK</b> .....	v
<b>ABSTRACT</b> .....	vii
<b>KATA PENGANTAR</b> .....	ix
<b>DAFTAR ISI</b> .....	xi
<b>DAFTAR GAMBAR</b> .....	xiii
<b>DAFTAR TABEL</b> .....	xiv

### BAB I PENDAHULUAN

1.1 Latar Belakang .....	1
1.2 Permasalahan .....	1
1.3 Batasan Masalah .....	2
1.4 Maksud dan Tujuan .....	2
1.5 Sistematika Laporan .....	2
1.6 Relevansi .....	3

### BAB II TEORI PENUNJANG

2.1 Pesawat <i>Aeromodeling FPV Raptor</i> .....	5
2.2 Cara Kerja Terbang Pesawat <i>Aeromodeling FPV Raptor</i> 11 .....	
2.3 Mikrokontroler AVR ATmega16 .....	13
2.3.1 Spesifikasi Mikrokontroler AVR ATmega16 .....	13
2.3.2 Konfigurasi <i>Pin</i> ATmega16 .....	13
2.4 Kamera <i>Wireless TX RX Boscam 5.8 GHz 200mW</i> .....	16
2.5 Op-Amp 358 .....	25
2.5.1 Konfigurasi Pin Op-Amp 358 .....	26
2.5.2 <i>Non Inverting</i> .....	27
2.6 <i>Filter RC</i> .....	28
2.7 RSSI ( <i>Received Signal Strength Indicator</i> ) .....	28

### BAB III PERANCANGAN DAN PEMBUATAN ALAT

3.1 Pesawat <i>Aeromodelling FPV Raptor 1600mm</i> .....	29
3.2 Rangkaian <i>Transmitter</i> dan <i>Receiver</i> Untuk Kendali Pesawat .....	30
3.3 Rangkaian <i>Transmitter</i> dan <i>Receiver</i> ke Monitor .....	31
3.4 Rangkaian Pergerakan Kamera .....	31
3.5 Perancangan Rangkaian Indikator Sinyal <i>Receiver</i> Video .....	33



3.6 Perancangan <i>Software</i> Pergerakan Kamera .....	35
3.7 Perancangan <i>Software</i> Sinyal Indikator Video .....	40
3.8 Perancangan penyangga remote control dan monitor .....	41
3.9 Sistem Secara Keseluruhan.....	43

#### **BAB IV PENGUJIAN DAN ANALISA DATA**

4.1 Pengujian Sumber Tegangan .....	45
4.2 Pengujian Sinyal <i>Transmitter</i> dan <i>Receiver</i> Video .....	46
4.3 Pengujian <i>Receiver Remote Control</i> .....	47
4.4 Pengujian pada Pergerakan Kamera .....	48
4.5 Pengujian Rangkaian Minimum Mikrokontroler .....	50
4.6 Pengujian LCD .....	51

#### **BAB V PENUTUP**

5.1 Kesimpulan.....	53
5.2 Saran .....	53

<b>DAFTAR PUSTAKA</b> .....	55
-----------------------------	----

#### **LAMPIRAN**

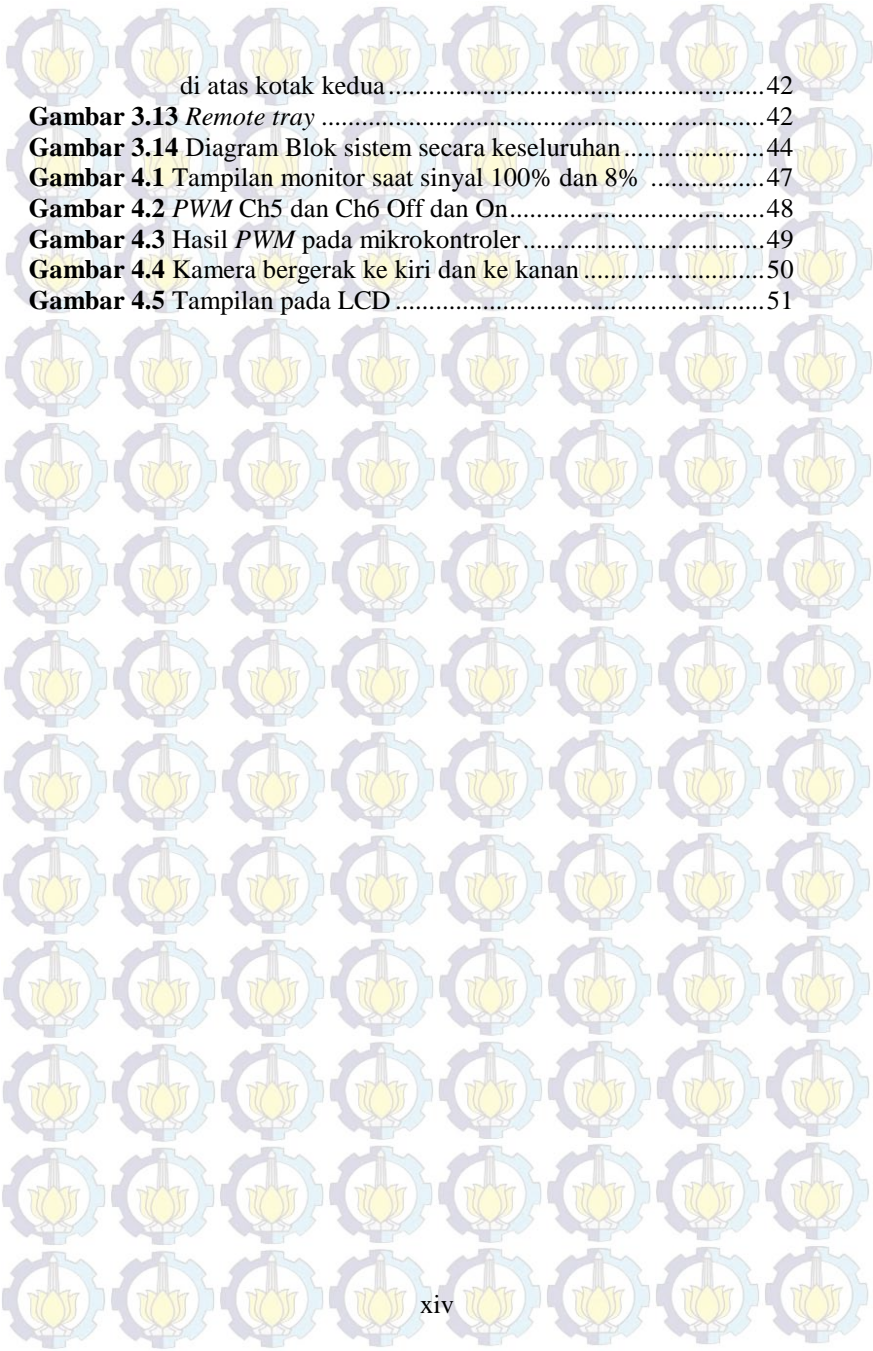
1. <i>Listing</i> Program <i>CodeWizardAVR</i> Pergerakan Kamera .....	A-1
2. <i>Listing</i> Program <i>CodeWizardAVR</i> Indikator Sinyal.....	B-1
3. Bentuk Fisik Keseluruhan Sistem .....	C-1
4. Bentuk Fisik Ground Station .....	D-1
5. Bentuk Fisik Mikrokontroler Untuk Pergerakan Kameradan Indikator Sinyal .....	E-1
6. Bentuk Fisik <i>OpAmp</i> Untuk Pergerakan Kamera dan Indikator Sinyal .....	F-1
7. Hasil Gambar Pemantauan .....	G-1
8. <i>Datasheet</i> ATMega16.....	H-1
9. <i>Datasheet</i> <i>OpAmp</i> 358.....	I-1

<b>RIWAYAT HIDUP</b> .....	J-1
----------------------------	-----



## DAFTAR GAMBAR

<b>Gambar 2.1</b> <i>Transmitter</i> Terpasang Dengan Remote .....	5
<b>Gambar 2.2</b> <i>Receiver</i> Turnigy XR7000 .....	6
<b>Gambar 2.3</b> <i>Micro Servo</i> 9g .....	7
<b>Gambar 2.4</b> Pulse With Modulation .....	8
<b>Gambar 2.5</b> Motor <i>Brushless</i> .....	8
<b>Gambar 2.6</b> Pesawat <i>FPV Raptor</i> .....	9
<b>Gambar 2.7</b> Arah Terbang Pesawat <i>Aeromodeling FPV Raptor</i> .....	12
<b>Gambar 2.8</b> Konfigurasi <i>Pin</i> Mikrokontroler ATmega16 .....	14
<b>Gambar 2.9</b> Rancangan Kamera <i>Wireless</i> .....	17
<b>Gambar 2.10</b> <i>Transmitter</i> Nirkabel Boscaml 5.8GHz 200mW .....	18
<b>Gambar 2.11</b> <i>Receiver</i> Nirkabel Boscaml 5.8GHz 200mW .....	19
<b>Gambar 2.12</b> <i>Hd Wing Camera</i> .....	21
<b>Gambar 2.13</b> Gen Ace 2200 MAH 3cell.....	23
<b>Gambar 2.14</b> Lpb Nano 2200 MAh 3cell.....	23
<b>Gambar 2.15</b> Black Fox 1300 MAh 2cell .....	24
<b>Gambar 2.16</b> Monitor <i>Rearview</i> 7in.....	25
<b>Gambar 2.17</b> LM 358.....	25
<b>Gambar 2.18</b> Konfigurasi <i>Pin OpAmp</i> 358 .....	26
<b>Gambar 2.19</b> Rangkaian <i>Non Inverting</i> .....	27
<b>Gambar 2.20</b> Rangkaian RSSI pada Boscaml 5.8 GHz 200mW .....	28
<b>Gambar 3.1</b> Ukuran Pesawat <i>FPV Raptor</i> 1600mm.....	30
<b>Gambar 3.2</b> Diagram Blok sistem <i>Transmitter</i> dan <i>Receiver</i> dalam pergerakan pesawat.....	30
<b>Gambar 3.3</b> Diagram Blok sistem <i>Transmitter</i> dan <i>Receiver</i> untuk penampilan .....	31
<b>Gambar 3.4</b> Diagram Blok Sistem kerja pergerakan kamera .....	32
<b>Gambar 3.5</b> Rangkaian <i>OpAmp</i> 358 untuk pergerakan kamera .....	32
<b>Gambar 3.6</b> Rangkaian Minimum Sistem AVR ATmega16 Pergerakan Kamera .....	33
<b>Gambar 3.7</b> Diagram Blok indikator sinyal <i>Receiver</i> video.....	34
<b>Gambar 3.8</b> Rangkaian <i>OpAmp</i> indikator sinyal .....	34
<b>Gambar 3.9</b> Rangkaian Minimum Sistem AVR ATmega16 Indikator Sinyal .....	35
<b>Gambar 3.10</b> Pengaturan timer untuk pergerakan servo .....	39
<b>Gambar 3.11</b> Algoritma indikator sinyal .....	40
<b>Gambar 3.12</b> a. Kotak pertama terletak di bawah, b. kotak kedua terletak di atas dan c. dudukan monitor terletak	



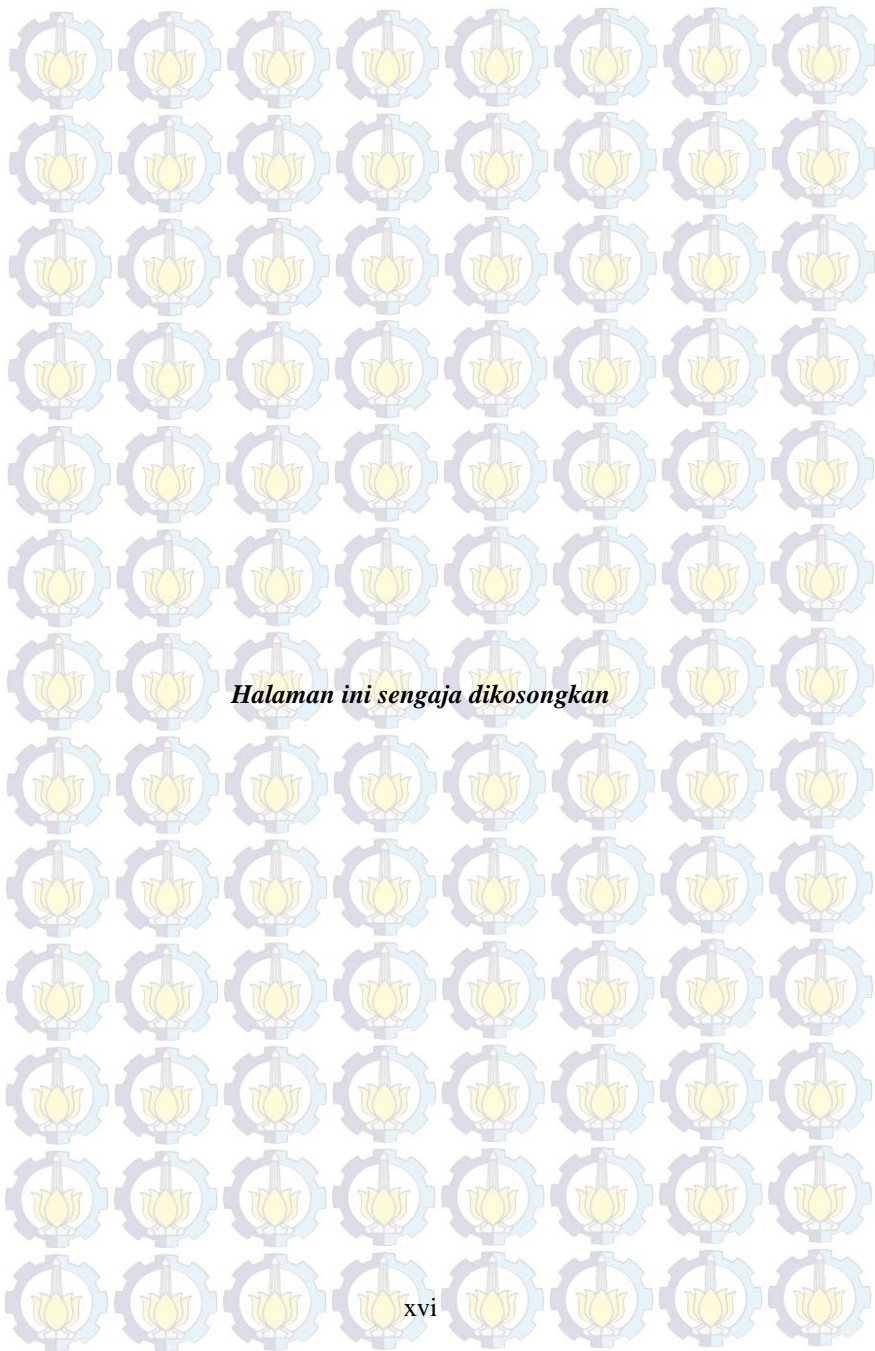
di atas kotak kedua .....	42
<b>Gambar 3.13</b> <i>Remote tray</i> .....	42
<b>Gambar 3.14</b> Diagram Blok sistem secara keseluruhan .....	44
<b>Gambar 4.1</b> Tampilan monitor saat sinyal 100% dan 8% .....	47
<b>Gambar 4.2</b> <i>PWM</i> Ch5 dan Ch6 Off dan On.....	48
<b>Gambar 4.3</b> Hasil <i>PWM</i> pada mikrokontroler .....	49
<b>Gambar 4.4</b> Kamera bergerak ke kiri dan ke kanan .....	50
<b>Gambar 4.5</b> Tampilan pada LCD .....	51



## DAFTAR TABEL

<b>Tabel 2.1</b>	Spesifikasi Turnigy 6X <i>Transmitter</i> .....	6
<b>Tabel 2.2</b>	Spesifikasi Turnigy XR7000 <i>Receiver</i> .....	6
<b>Tabel 2.3</b>	Spesifikasi Servo 9g .....	8
<b>Tabel 2.4</b>	Spesifikasi Motor Brushless .....	9
<b>Tabel 2.5</b>	Spesifikasi 30A <i>Brushless ESC</i> .....	9
<b>Tabel 2.6</b>	Spesifikasi awak pesawat <i>FPV Raptor</i> .....	10
<b>Tabel 2.7</b>	Fungsi khusus <i>Port A</i> .....	14
<b>Tabel 2.8</b>	Fungsi Khusus <i>Port C</i> .....	15
<b>Tabel 2.9</b>	Fungsi Khusus <i>Port D</i> .....	16
<b>Tabel 2.10</b>	Spesifikasi untuk pemancar kamera .....	18
<b>Tabel 2.11</b>	Frekuensi Kerja <i>Transmitter Boscaml</i> .....	19
<b>Tabel 2.12</b>	Spesifikasi <i>receiver</i> nirkabel 5.8GHz AV 200mW .....	20
<b>Tabel 2.13</b>	Frekuensi Kerja <i>Transmitter Boscaml</i> .....	20
<b>Tabel 2.14</b>	Spesifikasi <i>Hd wing camera</i> .....	21
<b>Tabel 2.15</b>	Spesifikasi Gen ace 2200 Mah 3cell .....	23
<b>Tabel 2.16</b>	Spesifikasi Lpb Nano 2200 Mah 3cell .....	23
<b>Tabel 2.17</b>	Spesifikasi Black Fox 1300 Mah 2cell .....	24
<b>Tabel 2.18</b>	Spesifikasi Monitor <i>rearview 7in</i> .....	25
<b>Tabel 4.1</b>	Pengujian sumber tegangan .....	45
<b>Tabel 4.2</b>	Pengujian <i>transmitter</i> dan <i>receiver</i> video saat terhalang ..	46
<b>Tabel 4.3</b>	Pengujian <i>transmitter</i> dan <i>receiver</i> video di lapangan saat tidak terhalang .....	47
<b>Tabel 4.4</b>	Pengujian Tegangan <i>Receiver remote control</i> .....	47
<b>Tabel 4.5</b>	Pengujian Jarak receiver pesawat .....	48
<b>Tabel 4.6</b>	Pengujian hasil output <i>Filter RC</i> .....	49
<b>Tabel 4.7</b>	Pengujian setelah penguatan 11 kali pada <i>OpAmp</i> .....	50
<b>Tabel 4.7</b>	Pengujian Kecepatan Pergerakan Kamera .....	50
<b>Tabel 4.9</b>	Pengujian Rangkaian Minimum .....	51





*Halaman ini sengaja dikosongkan*

# PENDAHULUAN

## BAB I

### 1.1. Latar Belakang

Seiring dengan perkembangan jaman dan perkembangan teknologi yang semakin cepat, tingkat kenyamanan, keamanan, dan kepraktisan diuntut semakin meningkat pula. Demikian halnya dengan penggunaan pesawat model yang semakin canggih. Pada umumnya pesawat FPV tidak memiliki fitur yang lengkap, hanya digunakan untuk kebutuhan hiburan dalam penerbangan. Sehingga pesawat tersebut kurang bermanfaat. Untuk memberikan fitur yang bermanfaat pada pesawat FPV, maka dibuatlah suatu inovasi sistem pergerakan kamera pada pesawat FPV, alat ini dimaksudkan untuk memudahkan survei udara yaitu memonitor kebakaran hutan, kemacetan lalu lintas, bencana banjir, dan patroli perbatasan.

Dengan adanya sistem pergerakan kamera pada pesawat FPV, diharapkan semua pengguna pesawat aeromodeling mendapatkan informasi hasil pantauan udara dengan mudah, yaitu dengan menggerakkan kamera sesuai dengan kebutuhan u. Juga dapat digunakan sebagai peralatan pengintai udara yang fleksibel bagi pengguna keamanan untuk memetakan wilayah patroli perbatasan yang kemungkinan terjadi tindak kriminal, seperti penebang pohon liar, pencurian ikan oleh negara lain.

### 1.2. Permasalahan

Hal yang melatar belakangi pembuatan sistem pergerakan kamera pada pesawat FPV adalah pesawat tersebut tidak memiliki fitur untuk menampilkan hasil penerbangan, bahkan jika ada fitur untuk kamera. Kamera tersebut tidak dapat dikontrol sesuai dengan keinginan untuk pengambilan video saat pesawat diterbangkan. Kemudian pesawat FPV hanya digunakan sebagai hiburan untuk penerbangan. Sehingga pesawat tersebut serasa kurang bermanfaat. Maka disinilah kegunaan sistem pergerakan kamera pada pesawat FPV yang dapat memudahkan dalam pemotretan survei udara terhadap suatu lokasi. Sebagai tambahan dan inovasi pada pesawat, Sistem pergerakan kamera tersebut diberi transmitter untuk mengirim hasil video saat kamera melakukan pergerakan. Serta *receiver* dan *monitor* untuk menampilkan hasil pantauan dari pergerakan. Sehingga pesawat tersebut memberikan banyak inovasi yang mudah digunakan untuk pemantauan udara secara



langsung, juga sangat memungkinkan untuk membantu memantau menjaga wilayah patroli perbatasan dengan fleksibel.

### **1.3. Batasan Masalah**

Berdasarkan permasalahan yang telah diuraikan di atas, batasan masalah dari tugas akhir ini adalah:

- a. Mikrokontroler yang digunakan dalam tugas akhir ini adalah mikrokontroler Atmega16 dengan bahasa pemrograman C.
- b. Pesawat yang digunakan adalah FPV raptor 1600mm x 1200 mm
- c. Menggunakan *remote control turnigy 6X*
- d. Menggunakan *reciever turnigy XR700*
- e. Kamera yang digunakan adalah *hd wing kamera*
- f. Monitor yang digunakan adalah *rearview 7in*
- g. Menggunakan lcd 16 x 2

### **1.4. Maksud dan Tujuan**

Tujuan kami menuliskan tugas akhir ini adalah:

- a. Merancang sebuah sistem pergerakan kamera pada pesawat FPV untuk memudahkan dalam pemotretan keadaan suatu wilayah.
- b. Menganalisa bagaimana kestabilan gambar video yang ditampilkan.
- c. Mempermudah masyarakat modern untuk memantau secara langsung keadaan darurat tanpa menggunakan helikopter.

### **1.5. Sistematika Laporan**

Pembahasan pada laporan Tugas Akhir ini terdiri dari lima bab, yaitu pendahuluan, teori penunjang, perencanaan dan pembuatan alat, pengujian dan analisa alat, serta penutup.

#### **Bab I Pendahuluan**

Membahas tentang latar belakang, permasalahan, batasan masalah, maksud dan tujuan, sistematika laporan, metodologi, serta relevansi.

#### **Bab II Teori Penunjang**

Menjelaskan teori penunjang yang dijadikan landasan dan mendukung dalam perencanaan dan pembuatan alat.



### Bab III Perencanaan dan Pembuatan Alat

Membahas perencanaan dan pembuatan perangkat keras yang meliputi rangkaian-rangkaian, desain mekanik serta miniatur tandon air, dan perangkat lunak yang meliputi program yang akan digunakan untuk mengaktifkan alat tersebut.

### Bab IV Pengujian dan Analisa Data

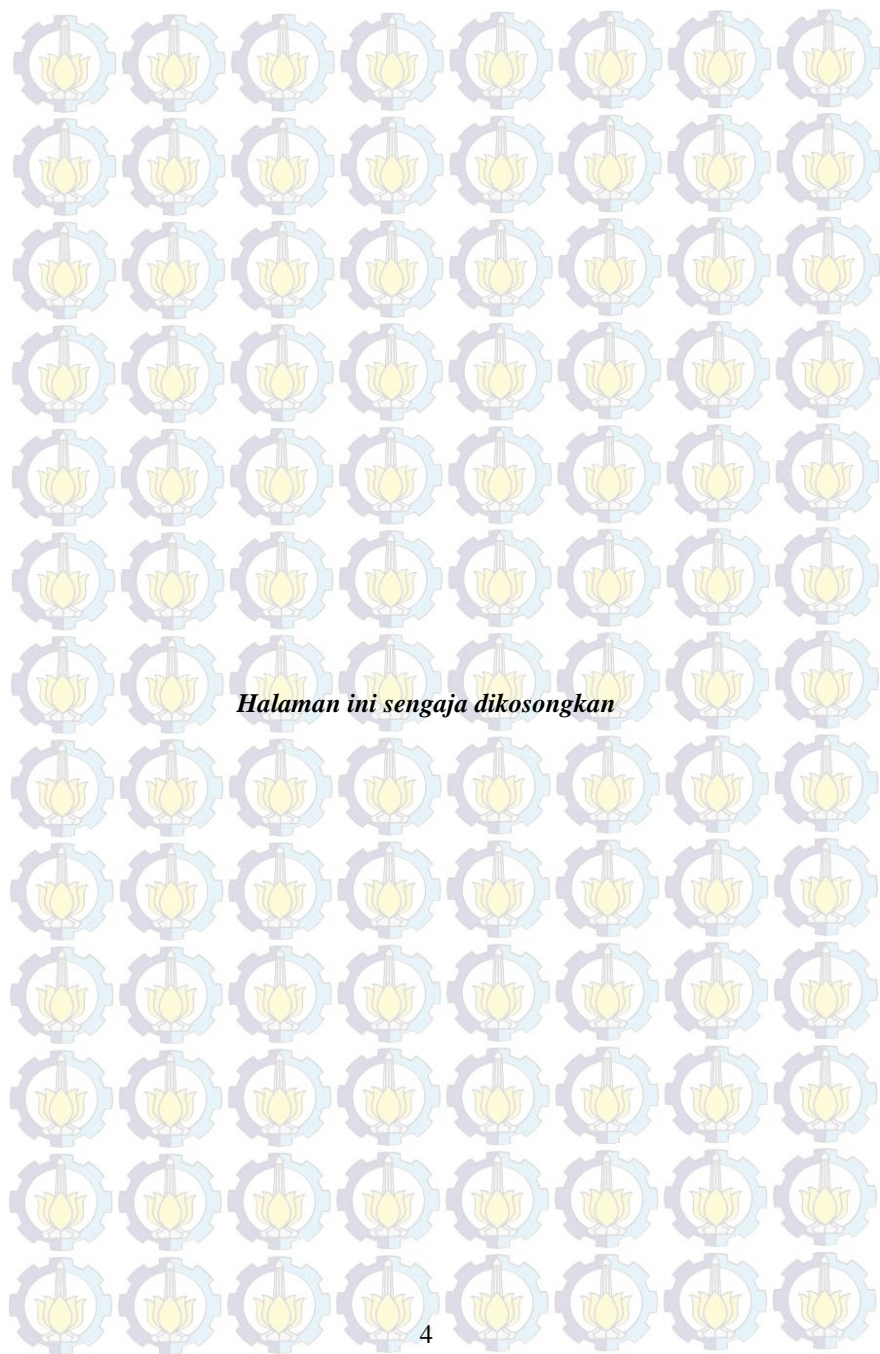
Membahas pengujian dan analisa data terhadap alat beserta sensor yang terdapat pada alat.

### Bab V Penutup

Berisi penutup yang menjelaskan tentang kesimpulan dari tugas akhir ini dan saran-saran untuk pengembangan alat ini lebih lanjut.

#### 1.7 Relevansi

Diharapkan dengan menyelesaikan tugas akhir ini kami bisa mendalami tentang bidang telemetri dalam hal perancangan pergerakan kamera pada pesawat model. Kemudian menjadi solusi untuk masyarakat luas dalam pengamatan daerah dari jarak jauh dengan wahana tanpa awak melalui udara dan diharapkan pesawat FPV ini mampu digunakan dalam kasus nyata di masyarakat.



## BAB II

### TEORI PENUNJANG

#### 2.1. Pesawat Aeromodeling FPV Raptor [ 1 ]

*Aeromodelling* adalah pesawat model yang lebih berat dari udara dengan ukuran-ukuran terbatas, baik bermotor dan tak dapat diawaki atau membawa manusia. Dengan adanya kata model dalam istilah *Aeromodelling*, dapat kita asumsikan bahwa pesawat-pesawat terbang yang digunakan oleh orang-orang yang berkecimpung dalam kegiatan aeromodelling ini tidak dapat dinaiki oleh manusia dan dalam hal pengendaliannya tidak secara langsung oleh pilot atau si pengendalinya seperti halnya pesawat sesungguhnya. Meskipun demikian kita tidak dapat beranggapan bahwa pesawat model ini hanya berukuran kecil saja akan tetapi pada kenyataannya ada pula pesawat model yang dibuat dengan ukuran yang mendekati ukuran sesungguhnya. Pesawat yang kami gunakan termasuk pesawat *model radio control*, yaitu pesawat model yang dikendalikan dengan sistem pengendali jarak jauh ( *Remote Control* ). Untuk nama pesawat ini adalah *FPV Raptor*, Pesawat tersebut memiliki bagian – bagian seperti berikut :

##### 1. Radio Control:

###### a. *Transmitter*



**Gambar 2.1** *Transmitter* Terpasang Dengan Remote

*Turnigy 6X Transmitter* adalah Pemancar besar dengan frekuensi 2.4 GHz yang menggunakan teknologi handal. *Turnigy 6X Transmitter* memiliki kualitas yang bagus terhadap tampilannya, terutama mengingat harganya yang murah, fiturnya menarik dan saklar tertutup sebagai pembalik arah servo.



Spesifikasi *Turnigy 6X Transmitter*:

Resolusi	1024
Frekuensi	2.4GHz ISM
Modulasi	GFSK
Spread Spectrum Mode	FHSS
Jumlah Frekuensi Saluran	20
Output Power	$\leq 20\text{dBm}$
Bekerja saat ini	$\leq 150\text{mA}$
Bekerja Voltage	1.2V x 4 (AA / NiMH)
Dimensi	200mm x 185mm x 105mm

**Tabel 2.1** Spesifikasi *Turnigy 6X Transmitter*

b. *Receiver*



**Gambar 2.2** Receiver *Turnigy XR7000*

*Receiver Turnigy XR7000* adalah penerima besar dengan frekuensi 2.4 GHz yang menggunakan teknologi handal. *Receiver Turnigy XR7000* memiliki kualitas yang bagus dengan memberikan kelebihan 7 channel. Sehingga dapat digunakan sesuai kebutuhan kontrol.

Spesifikasi *Turnigy XR7000 Receiver*:

Channel	7 (inc input RX Battery)
Frekuensi	2.4G Rentang Frekuensi ISM
Power	4.5v ~ 9.6v / $<30\text{mA}$
Berat Bersih	11.5g

Dimensi

41mm x 28mm x 14mm

**Tabel 2.2** Spesifikasi *Turnigy XR7000 Receiver*

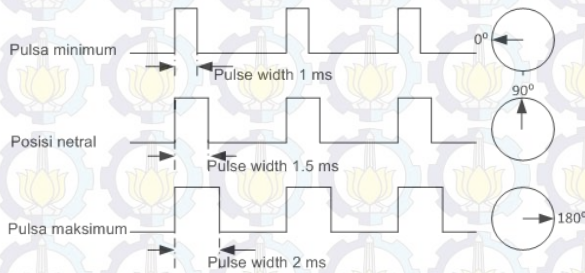
c. servo [ 6 ]



**Gambar 2.3** *Micro Servo 9g*

Motor servo adalah sebuah perangkat atau aktuatur putar (motor) yang dirancang dengan sistem kontrol umpan balik loop tertutup (servo), sehingga dapat di set-up atau di atur untuk menentukan dan memastikan posisi sudut dari poros output motor. motor servo merupakan perangkat yang terdiri dari motor DC, serangkaian gear, rangkaian kontrol dan potensiometer. Serangkaian gear yang melekat pada poros motor DC akan memperlambat putaran poros dan meningkatkan torsi motor servo, sedangkan potensiometer dengan perubahan resistansinya saat motor berputar berfungsi sebagai penentu batas posisi putaran poros motor servo.

Prinsip kerja motor servo dikendalikan dengan memberikan sinyal modulasi lebar pulsa (Pulse Wide Modulation / PWM) melalui kabel kontrol. Lebar pulsa sinyal kontrol yang diberikan akan menentukan posisi sudut putaran dari poros motor servo. Sebagai contoh, lebar pulsa dengan waktu 1,5 ms (mili detik) akan memutar poros motor servo ke posisi sudut  $90^{\circ}$ . Bila pulsa lebih pendek dari 1,5 ms maka akan berputar ke arah posisi  $0^{\circ}$  atau ke kiri (berlawanan dengan arah jarum jam), sedangkan bila pulsa yang diberikan lebih lama dari 1,5 ms maka poros motor servo akan berputar ke arah posisi  $180^{\circ}$  atau ke kanan (searah jarum jam). Lebih jelasnya perhatikan Gambar 2.4.



**Gambar 2.4** Pulse With Modulation

Servo mikro 9g Tower Pro dengan torsi 1.5kg/cm adalah pilihan yang baik untuk pesawat terbang RC. Memberikan keuntungan dengan lengan servo dan sekrup untuk kemudahan instalasi. Spesifikasi servo 9g.

Dimensi	30.20x12.25x29.05mm
Operasi Kecepatan	0.3sec/60degree (4.8V)
Tegangan Operasi	4.2-6V
Rentang Suhu	0 ° -55 ° C
Berat	9g

**Tabel 2.3** Spesifikasi Servo 9g

#### d. Motor *Brushless*



**Gambar 2.5** Motor *Brushless*

Motor brushless adalah motor sinkron yang didukung oleh sumber listrik DC. Motor brushless tidak mengalami Slip, tidak seperti yang terjadi pada motor induksi biasa. Motor jenis ini mempunyai permanen magnet pada bagian rotor sedangkan elektro-magnet pada bagian stator. Artinya medan magnet yang dihasilkan oleh stator dan medan magnet



yang dihasilkan oleh rotor berputar di frekuensi yang sama. Setelah itu, dengan menggunakan sebuah rangkaian sederhana (simpler computer system) atau menggunakan ESC (Electronic Speed Control), maka kita dapat merubah arus di elektro-magnet ketika bagian rotor berputar, Sehingga dapat mengatur kecepatan motor.

Spesifikasi untuk motor brushless :

Nama Motor	Volantex
Max Efisiensi	80%
Efisiensi tinggi	4-10A (> 75%)
Kapasitas saat ini	12A/60s
Jumlah sel	2-3 Li-Poly

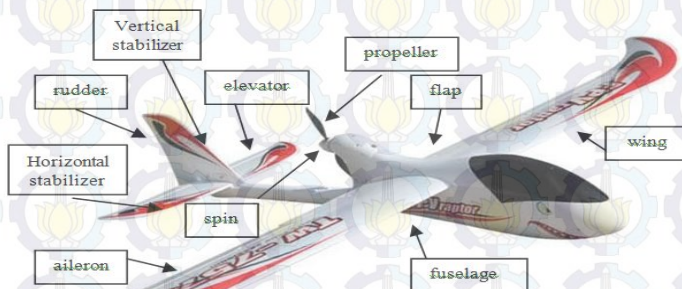
**Tabel 2.4** Spesifikasi Untuk Motor Brushless

Spesifikasi untuk 30A *Brushless ESC Volantex speed controller*:

Output	30A terus-menerus, meledak 40A dalam 10 detik.
Input Voltage	2-4 baterai lithium atau 5-12 sel NiCd / NiMH.
BEC	5V/1A (mode linear).

**Tabel 2.5** Spesifikasi Untuk 30A *Brushless ESC*

## 2. Pesawat FPV Raptor



**Gambar 2.6** Pesawat FPV Raptor

Spesifikasi awak pesawat *FPV Raptor* :

Lebar sayap	1600mm
Panjang	1044mm
Terbang Berat	950g
Motor	2812 1400kv brushless lari cepat
ESC	30A w / BEC
Propeller	8x4/203x102
Servo	9g x 4

**Tabel 2.6** Spesifikasi awak pesawat *FPV Raptor*

a. Body (*Fuselage*)

Adalah badan pesawat yang memiliki banyak fungsi. Disamping tempat tempel berbagai komponen utama lain badan pesawat juga terdapat kokpit juga kabin yang berisi bangku-bangku penumpang dan juga bagasi yang dapat diganti dengan bangku tambahan

b. Sayap (*Wing*)

Ketika aliran udara melewati sayap pesawat, aliran ini menyebabkan gaya angkat yang membuat pesawat dapat terbang. Sayap dibentuk untuk mengambil manfaat gaya ini secara maksimal. Ada yang terletak di atas, tengah maupun dibawah badan pesawat.

c. *Aileron*

*Aileron* (kemudi guling) terbentang dari tengah sayap sampai keujung tiap sayap yang bergerak keatas dan kebawah secara berlawanan pada masing-masing sayap, dengan membelokkan seperti stir mobil biasa. Jika *aileron* kanan naik maka yang kiri turun juga sebaliknya.

d. *Flap*

*Flap* terbentang mulai pangkal sampai pertengahan sayap ditiap sayap, *flap* ini selalu bergerak searah, jika *flap* kiri turun maka *flap* kanan juga turun dengan sudut yang sama. Dikendalikan dengan menarik tuas seperti rem tangan pada mobil, atau memutar roda kerekan atau juga secara elektris pada tipe pesawat yang lebih muda tahun pembuatannya.



e. *Rudder*

Terletak pada bagian belakang bersamaan dengan *vertical stabilizer* yang berfungsi sebagai pengarah belokan pesawat ke kanan ataupun ke kiri.

f. *elevator*

Terletak pada bagian belakang bersama dengan *horizontal stabilizer* yang berfungsi sebagai arah belokan pesawat arah ke bawah ataupun ke atas.

g. *Empennage* / ekor

Pada ekor pesawat Masing-masing tak bergerak dan berfungsi seperti bulu pada panah untuk menstabilkan pesawat dan membantu Anda meluruskan pesawat.

h. *propeller*

Baling – baling yang berguna untuk menghasilkan dorongan terhadap pesawat untuk melakukan penerbangan yang dibantu dengan adanya *spin* sebagai pemutar baling – baling yang digunakan pesawat.

## 2.2. Cara Kerja Terbang Pesawat Aeromodeling FPV Raptor [ 2 ]

Seperti kita ketahui bahwa bumi kita di selimuti oleh lapisan udara yang disebut atmosfer. Dan dapat dimengerti setiap benda yang memiliki berat dapat jatuh ke bumi. Lalu bagaimana dengan pesawat udara yang beratnya berton-ton dapat terbang juga pesawat aeromodeling yang lumayan berat bisa terbang. Berikut adalah cara yang realistis untuk mengetahui pesawat bisa terbang diudara.

Untuk memahami hal ini kita harus mengerti bahwa ada 4 gaya yang bekerja pada pesawat udara selama penerbangan yaitu Gaya angkat ( *lift* ) atau gaya keatas, Gaya berat ( *weight* ) atau gaya kebawah, selanjutnya gaya maju ( *thrust* ) serta gaya kebelakang ( *drag* ). Dua gaya berikut dapat mudah dipahami. Gaya berat ( *weight* ) bekerja menarik benda kembali ke bumi, sebagai contoh apabila kita melemparkan batu ke atas maka akan jatuh. Selanjutnya apabila kita mengendarai sepeda, maka terasa hambatan dari depan. Berikut gambar arah terbang pesawat :





**Gambar 2.7** Arah Terbang Pesawat *Aeromodeling FPV Raptor*

Pada dasarnya pesawat terbang adalah benda yg lebih berat dari udara yg melayang diudara karena memanfaatkan efek aerodinamis di udara. sebagai gambaran dibawah ada ilustrasi sedikit mengenai gaya, dg bahasa yg sederhana, pergerakan yg terjadi pada pesawat terbang dan juga geometri nya.

1. Dimulai dengan gaya – gaya yang terjadi/bekerja pada pesawat, yaitu:

- a. *Lift*, gaya angkat yg terjadi karena benda yg aerodinamis bergerak diudara.
- b. *Weight*, gaya berat karena pengaruh tarikan gravitasi bumi
- c. *Thrust* , gaya dorong karena adanya mesin yg menggerakkan pesawat d. *Drag*, gaya hambat yg terjadi karena ada benda padat bergerak diudara.

2. Pergerakan pesawat diudara. Pada dasarnya pesawat terbang mempunyai 3 sumbu pergerakan (x/y/z axis) yg disebut :

- a. *Roll* , pergerakan pesawat terhadap sumbu horisontal depan belakang yg mengakibatkan pesawat berguling kiri kanan (badan pesawat diam, sayap kiri kanan yg turun naik)
- b. *Yaw*, pergerakan pesawat terhadap sumbu vertikal yg menyebabkan hidung pesawat berubah arah kiri kanan (pesawat akan berbelok kiri kanan).
- c. *Pitch*, pergerakan pesawat terhadap sumbu horisontal yg tegak lurus terhadap sumbu roll yg menyebabkan hidung pesawat akan turun atau naik.

## 2.3 Mikrokontroler AVR ATmega16 [ 3 ]

Mikrokontroler AVR (*Alf and Vegard's Risc processor*) merupakan pengontrol utama standar industri dan riset saat ini. Hal ini dikarenakan berbagai kelebihan yang dimilikinya dibandingkan dengan mikroprosesor, yaitu murah, dukungan software dan dokumentasi yang memadai, dan membutuhkan komponen pendukung yang sangat sedikit. Salah satu tipe mikrokontroler AVR untuk aplikasi standar yang memiliki fitur memuaskan ialah ATmega16.

Mikrokontroler atau mikroprosesor adalah suatu piranti yang digunakan untuk mengolah data-data biner (*digital*) yang di dalamnya merupakan gabungan dari rangkaian-rangkaian elektronik yang dikemas dalam bentuk suatu *chip* (IC). Pada umumnya mikrokontroler terdiri dari bagian-bagian sebagai berikut: alamat (*address*), data, pengendali, memori (*RAM atau ROM*), dan bagian *input-output*.

### 2.3.1. Spesifikasi Mikrokontroler AVR ATmega16

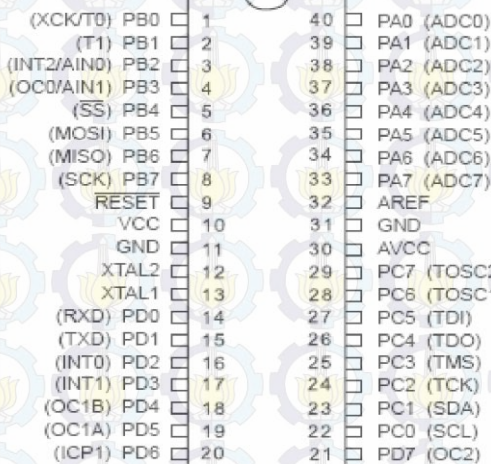
Fitur-fitur pada mikrokontroler ATmega16 antara lain:

- a. Saluran I/O ada 32 buah, yaitu *Port A*, *Port B*, *Port C*, dan *Port D*.
- b. ADC 10 bit sebanyak 8 *channel*.
- c. CPU yang terdiri dari 32 buah *register*.
- d. *Watchdog Timer* dengan osilator *internal*.
- e. Dan fitur-fitur lain yang mempermudah dalam penggunaan.
- f. Tegangan kerja berkisar 4-5 V.
- g. Memori Flash 8 *Kbytes* untuk program
- h. Memori EEPROM 512 *bytes* untuk data
- i. Memori SRAM 512 *bytes* untuk data
- j. 20 *interrupt*
- k. Satu 16-bit *timer* dan dua 8-bit *timer*
- l. Komunikasi serial melalui SPI dan USART
- m. *Analog komparator*
- n. 4 I/O PWM
- o. Fasilitas *In System Programming (ISP)*

### 2.3.2. Konfigurasi Pin ATmega16

Gambar di bawah ini merupakan susunan kaki standar 40 *Pin* Dari DIP mikrokontroler AVR ATmega16. Berikut ini adalah penjelasan dari beberapa pin yang digunakan dalam rangkaian yang telah dibuat.





(XCK/T0) PB0	1	40	PA0 (ADC0)
(T1) PB1	2	39	PA1 (ADC1)
(INT2/AIN0) PB2	3	38	PA2 (ADC2)
(OC0/AIN1) PB3	4	37	PA3 (ADC3)
(SS) PB4	5	36	PA4 (ADC4)
(MOSI) PB5	6	35	PA5 (ADC5)
(MISO) PB6	7	34	PA6 (ADC6)
(SCK) PB7	8	33	PA7 (ADC7)
RESET	9	32	AREF
VCC	10	31	GND
GND	11	30	AVCC
XTAL2	12	29	PC7 (TOSC2)
XTAL1	13	28	PC6 (TOSC1)
(RXD) PD0	14	27	PC5 (TDI)
(TXD) PD1	15	26	PC4 (TDO)
(INT0) PD2	16	25	PC3 (TMS)
(INT1) PD3	17	24	PC2 (TCK)
(OC1B) PD4	18	23	PC1 (SDA)
(OC1A) PD5	19	22	PC0 (SCL)
(ICP1) PD6	20	21	PD7 (OC2)

**Gambar 2.8** Konfigurasi *Pin* Mikrokontroler ATmega16

- VCC merupakan *Pin* masukan positif catu daya.
- GND sebagai *Pin* ground.

Port Pin	Fungsi Khusus
<b>PA0</b>	ADC0 (ADC input channel 0)
<b>PA1</b>	ADC1 (ADC input channel 1)
<b>PA2</b>	ADC2 (ADC input channel 2)
<b>PA3</b>	ADC3 (ADC input channel 3)
<b>PA4</b>	ADC4 (ADC input channel 4)
<b>PA5</b>	ADC5 (ADC input channel 5)
<b>PA6</b>	ADC6 (ADC input channel 6)
<b>PA7</b>	ADC7 (ADC input channel 7)

**Tabel 2.7** Fungsi khusus *Port A*

- Port A* (PA0...PA7) merupakan *Pin* I/O dua arah dan dapat diprogram sebagai *Pin* masukan ADC. Merupakan 8-bit *directional port* I/O. Setiap *pin*nya dapat menyediakan *internal pull-up resistor* (dapat diatur per bit). *Output buffer Port A* dapat memberi arus 20 mA dan dapat mengendalikan *display*



LED secara langsung. Data *direction register port A* (DDRA) harus diatur terlebih dahulu sebelum *port A* digunakan. Bit-bit DDRA diisi 0 jika ingin memfungsikan *pin-pin port A* yang bersesuaian sebagai *input*, atau diisi 1 jika sebagai *output*. Selain itu, kedelapan *pin port A* juga digunakan untuk masukan sinyal analog bagi A/D converter.

Port Pin	Fungsi Khusus
PC0	SCL (Two-wire Serial Bus Clock Line)
PC1	SDA (Two-wire Serial Bus Data Input/Output Line)
PC2	TCK (JTAG Test Clock)
PC3	TMS (JTAG Test Mode Select)
PC4	TDO (JTAG Test Data Out)
PC5	TDI (JTAG Test Data In)
PC6	TOSC1 (Timer Oscillator Pin 1)
PC7	TOSC2 (Timer Oscillator Pin 2)

**Tabel 2.8** Fungsi Khusus *Port C*

- d. *Port C* (PC0...PC7) merupakan *Pin I/O* dua arah dengan fungsi khusus yaitu TWI, komparator *Analog*, dan *Timer Oscillator*. Merupakan 8-bit directional *port I/O*. Setiap *pinnya* dapat menyediakan internal *pull-up* resistor (dapat diatur per bit). *Output buffer Port C* dapat memberi arus 20 mA dan dapat mengendalikan *display LED* secara langsung. *Data Direction Register port C* (DDRC) harus diatur terlebih dahulu sebelum *Port C* digunakan. Bit-bit DDRC diisi 0 jika ingin memfungsikan *pin-pin port C* yang bersesuaian sebagai *input*, atau diisi 1 jika sebagai *output*. Selain itu, dua *pin port C* (PC6 dan PC7) juga memiliki fungsi sebagai *oscillator* untuk *timer/counter*.

Port Pin	Fungsi Khusus
PD0	RDX (UART <i>input line</i> )
PD1	TDX (UART <i>output line</i> )
PD2	INT0 ( <i>external interrupt 0 input</i> )
PD3	INT1 ( <i>external interrupt 1 input</i> )
PD4	OC1B ( <i>Timer/Counter1 output compareB match output</i> )

<b>PD5</b>	OC1A ( <i>Timer/Counter1 output compareA match output</i> )
<b>PD6</b>	ICP ( <i>Timer/Counter1 input capture pin</i> )
<b>PD7</b>	OC2 ( <i>Timer/Counter2 output compare match output</i> )

**Tabel 2.9** Fungsi Khusus *Port D*

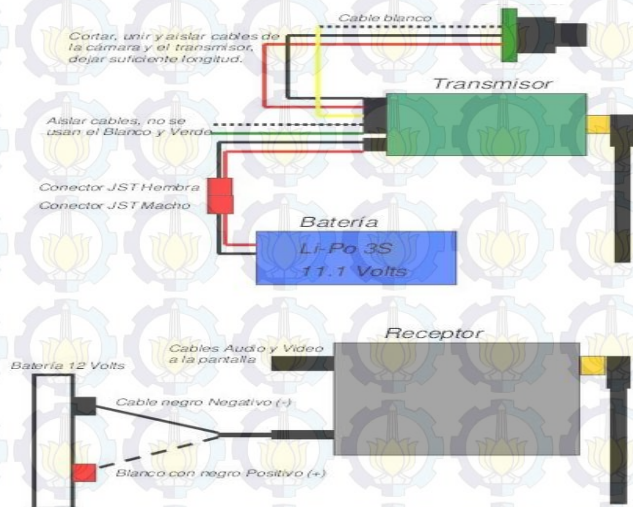
- e. *Port D* (PD0...PD7) merupakan *Pin I/O* dua arah dengan fungsi khusus yaitu komparator *Analog*, interupsi *eksternal*, dan komunikasi serial. Merupakan 8-bit *directional port I/O*. Setiap *pin*nya dapat menyediakan *internal pull-up* resistor (dapat diatur per bit). *Output buffer Port D* dapat memberi arus 20 mA dan dapat mengendalikan *display LED* secara langsung. *Data Direction Register port D* (DDRD) harus diatur terlebih dahulu sebelum *Port D* digunakan. Bit-bit DDRD diisi 0 jika ingin memfungsikan *pin-pin port D* yang bersesuaian sebagai *input*, atau diisi 1 jika sebagai *output*. Selain itu, *pin-pin port D* juga memiliki untuk fungsi khusus seperti yang dapat dilihat dalam Tabel 2.9.
- f. *Reset* merupakan *pin* yang digunakan untuk me-reset mikrokontroler. RST pada *pin* 9 merupakan reset dari AVR. Jika pada *pin* ini diberi masukan *low* selama minimal 2 *machine cycle* maka system akan di-reset.
- g. XTAL1 dan XTAL2 sebagai *Pin* masukan *clock eksternal*. XTAL1 adalah masukan ke *inverting oscillator amplifier* dan *input* ke *internal clock operating circuit*. XTAL2 adalah *output* dari *inverting oscillator amplifier*.
- h. AVCC sebagai *Pin* masukan tegangan untuk ADC. Kaki ini harus secara *eksternal* terhubung ke Vcc melalui *lowpass filter*.
- i. AREF sebagai *Pin* masukan tegangan referensi bagi A/D Converter. Untuk operasionalisasi ADC, suatu *level* tegangan antara AGND dan Avcc harus diberikan ke kaki ini. AGND adalah kaki untuk *analog ground*. Hubungkan kaki ini ke GND, kecuali jika *board* memiliki *analog ground* yang terpisah.

## 2.4 Kamera Wirreles TX RX Boscam 5.8G 200mW [ 4 ]

Seperti yang saya ketahui Tx dan Rx adalah sebuah protokol yang diimplementasikan dalam sebuah perangkat bernama UART (*Universal Asynchronous Receiver / Transmitter*). Rx adalah jalur penerimaan data



(perpindahan data) dari satu komputer ke komputer lain. Rx biasa disebut received, yang berguna menangkap data yang dikirim oleh transmitter (Tx). Tx disebut transmitter yang berfungsi untuk mengirim data/mengeluarkan data, atau merupakan jalan yang dilalui dalam mengirim data antar device. data akan dikirim melalui Tx (transmitter) dan di ujung lainnya data akan diterima melalui Rx (Received). Untuk menampilkan video pada monitor maka dibutuhkan TX dan RX. Awalnya TX dihubungkan dengan kamera HD wing, Kemudian RX dihubungkan dengan monitor untuk menampilkan hasil video yang akan ditampilkan, Selanjutnya RX dipasang terpisah dengan TX sesuai dengan kebutuhan, Untuk lebih jelasnya dalam perancangan kamera sesuai dengan Gambar 2.9.



**Gambar 2.9** Rancangan Kamera Wireless

a. *Transmitter Nirkabel Boscam 5.8Ghz 200mW*

*Transmitter nirkabel boscam 5.8Ghz 200mW* mampu mengirim data sejauh  $\pm 500$  m saat terbang dengan antenna yang disertakan. Selanjutnya sebuah antenna dengan opsi langsung dapat memperpanjang rentang pengiriman data sejauh 1km +. Unit penerima memiliki output AV tambahan untuk merekam video saat menggunakan *HD wing camera*.





**Gambar 2.10** Transmitter Nirkabel Boscama 5.8Ghz 200mW

Spesifikasi untuk pemancar kamera :

Merek Nama	Boscama
Pemancar frekuensi	5645 - 5945MHz ; 8CH
Frekuensi Audio	6.5MHz
Level input Video	0.8 ~ 1.2Vp - p
Video impedansi masukan	75Ohm
Tingkat input Audio	0,5 ~ 2.0Vp - p
Audio impedansi masukan	10K/Ohm
Berat	25g
Konektor ANT	SMA (di dalam jarum )
Video Output	1Vp - p 75Ω
Audio output	1Vp - p 600Ω
Power supply	DC 12V
Video	PAL / NTSC
Tenaga Keluaran	21~23dBm
Dimensi	5.5mm x 68mm x 13mm

**Tabel 2.10** Spesifikasi untuk pemancar kamera

Frekuensi Kerja Transmitter Boscam :

CH1	5705
CH2	5685
CH3	5665
CH4	5645
CH5	5885
CH6	5905
CH7	5925
CH8	5945

**Tabel 2.11** Frekuensi Kerja Transmitter Boscam

b. *Receiver Nirkabel Boscam 5.8G 200mW*



**Gambar 2.11** *Receiver Nirkabel Boscam 5.8G 200mW*

*Receiver Nirkabel Boscam 5.8G AV 200mW* mampu menerima data sejauh  $\pm 500$  m saat terbang dengan antenna yang disertakan. Selanjutnya sebuah antenna dengan opsi langsung dapat memperpanjang rentang penerimaan data sejauh 1km +.

**Spesifikasi Receiver Nirkabel Boscam 5.8G AV 200mW**

Merek Nama	Boscam
Channels	8
Tegangan operasi Receiver	5V - 12V DC
Suhu operasi	-10 ~ 85C
Bandwidth Video	0 ~ 8.0MHz
Frekuensi Audio	6.5MHz
Level input Video	0.8 ~ 1.2Vp-p
Video impedansi masukan	75Ohm
Tingkat input Audio	0,5 ~ 2.0Vp-p
Audio impedansi masukan	10K/Ohm
Konektor antenna	SMA
Dimensi	7cm * 5.2cm * 1.3cm
Sensitivitas	-90dbm
Berat	110g

**Tabel 2.12** Spesifikasi receiver nirkabel 5.8G AV 200mW

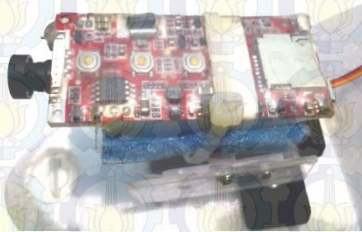
Frekuensi Kerja Transmitter Boscam :

CH1	5705
CH2	5685
CH3	5665
CH4	5645
CH5	5885
CH6	5905
CH7	5925
CH8	5945

**Tabel 2.13** Frekuensi Kerja Transmitter Boscam



c. *Hd Wing Camera* [ 5 ]



**Gambar 2.12** *Hd Wing Camera*

*Hd Wing Camera* menawarkan video yang berkualitas HD. bentuknya sangat kecil dan ringan . Dengan dukungan hingga 32GB kartu Transflash (Micro SD ) dapat menembak jam rekaman HD 720p . Selanjutnya dengan menggunakan baterai 850mAh Lipoly menyediakan [hingga 3 jam waktu perekaman . Kamera ini cocok untuk sistem FPV , Dapat melakukan monitoring secara langsung untuk menangkap penerbangan hasil penerbangan. Bahkan jika tidak menjadi kamera pada pesawat, Sangat bagus untuk digunakan dalam model apapun . dengan kualitas HD sesuai dengan kebutuhan pemakai.

Spesifikasi *Hd wing camera* :

lensa kamera HD	5 Megapixels
Resolusi Video	1280x720p 30fps
Kualitas video Dipilih	Standard / Superior / ekonomi
mendeteksi rekaman	10 detik , 1min , 5min , 10min
Resolusi Foto	2560x1920 JPG
Rekaman	menyediakan hingga 3 jam rekaman
Slot Memory	Transflash ( Micro SD ) hingga 32 GB
Ukuran	74mm x 28mm x 15mm
Berat	29.4g
Mode TV - OUT	PAL / NTSC

**Tabel 2.14** Spesifikasi *Hd wing camera*

#### d. Baterai Lipo

Baterai Lithium Polimer atau biasa disebut dengan LiPo merupakan salah satu jenis baterai yang sering digunakan dalam dunia RC. Utamanya untuk RC tipe pesawat. Baterai LiPo tidak menggunakan cairan sebagai elektrolit melainkan menggunakan elektrolit polimer kering yang berbentuk seperti lapisan plastik film tipis. Lapisan film ini disusun berlapis-lapis diantara anoda dan katoda yang mengakibatkan pertukaran ion. Dengan metode ini baterai LiPo dapat dibuat dalam berbagai bentuk dan ukuran. Diluar dari kelebihan arsitektur baterai LiPo, terdapat juga kekurangan yaitu lemahnya aliran pertukaran ion yang terjadi melalui elektrolit polimer kering. Hal ini menyebabkan penurunan pada charging dan discharging rate. Masalah ini sebenarnya bisa diatasi dengan memanaskan bat erai sehingga menyebabkan pertukaran ion menjadi lebih cepat, namun metode ini dianggap tidak dapat untuk diaplikasikan pada keadaan sehari-hari. Seandainya para ilmuwan dapat memecahkan masalah ini maka risiko keamanan pada batera jenis lithium akan sangat berkurang.

Beberapa kelebihan dari baterai LiPo adalah :

- a. Discharge rate lebih tinggi sehingga memungkinkan untuk menghasilkan daya tahan baterai yang lebih tinggi.
- b. LiPo tidak memiliki memori sehingga anda tidak perlu selalu mengosongkannya sebelum mengisi ulang.
- c. LiPo terbuat dari materi sejenis gel yang secara teori jauh lebih ringan.

Kekurangan baterai LiPo adalah :

- a. LiPo hanya bisa diisi ulang menggunakan charger yang mendukung type Lithium Polymer.
- b. Kemungkinan kerusakan permanen jika tegangan sel menurun di bawah 3V.
- c. Setelah 50 kali pengisian ulang, kapasitasnya bisa menurun 50-80% dari kapasitas pada saat masih baru.
- d. Harganya relative lebih mahal dibandingkan tipe lainnya.
- e. Mudah terbakar (mengejar api), salah satu kemungkinannya apabila terjadi overcharge.

Baterai Lipo yang digunakan :

1. Gen ace 2200 Mah 3cell

2. Lpb nano 2200 Mah 3cell
3. Black Fox 1300 Mah 2cell
1. *Gen ace 2200 Mah 3cell*



**Gambar 2.13** *Gen Ace 2200 Mah 3cell*

Spesifikasi *Gen ace 2200 Mah 3cell* :

Jenis Produk	lipo baterai
Kapasitas	2200mAh
Voltage	11.1V
<i>Max Discharge</i>	55C (121A)
<i>Max Burst Discharge</i>	110C (242A)
Berat	184g
Dimensi	105,05 * 33,53 * 24.27mm
Discharge Plug	steker T
Mengisi Rate	1-3C direkomendasikan, 5C Max

**Tabel 2.15** Spesifikasi *Gen ace 2200 Mah 3cell*

2. *Lpb nano 2200 Mah 3cell*



**Gambar 2.14** *Lpb Nano 2200 Mah 3cell*



Spesifikasi *Lpb Nano 2200 Mah 3cell* :

Kapasitas	2200mAh
Voltage	3S1P / 3 Cell / 11.1V
Discharge	25C
Berat	300g
Dimensi	115x34x24mm
Balance Plug	JST-XH
Discharge Plug	XT60

**Tabel 2.16** Spesifikasi *Lpb Nano 2200 Mah 3cell*

### 3. *Black Fox 1300 Mah 2cell*



**Gambar 2.15** *Black Fox 1300 Mah 2cell*

Spesifikasi *Black Fox 1300 Mah 2cell* :

Kapasitas	1300mAh
Voltage	2S1P / 2 Cell / 7.4V
Discharge	25C
Berat	81g
Dimensi	67x33x10mm
Balance Plug	JST-XH
Discharge Plug	JST

**Tabel 2.17** Spesifikasi *Black Fox 1300 Mah 2cell*

e. *Monitor Rearview 7in*

Monitor adalah alat yang digunakan untuk menampilkan gambar ataupun video. Untuk menampilkan hasil pemantauan secara langsung menggunakan monitor 7 in yang mudah dipasang dan lebih mudah dalam penggunaanya.



**Gambar 2.16** *Monitor Rearview 7in*

Spesifikasi *Monitor rearview 7in* :

Ukuran layar	7 inch (16:9)
Gelar Resolusi	480 (W) x RGB x 234 (H)
Sistem	PAL / NTSC
Power supply	7W, DC 12V
Resolusi Horizontal	500 TV Garis
Video	input video dua arah
Bluetooth Versi	1.2
Profil	Headset, A2DP
Kelas	2
Rentang Operasi	10 Meter.

**Tabel 2.18** Spesifikasi *Monitor rearview 7in*

## 2.5 Op-Amp 358



**Gambar 2.17** LM 358

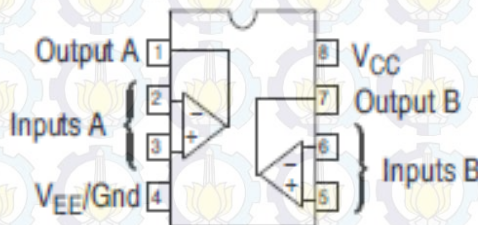
Penguat operasional adalah suatu rangkaian elektronika yang dikemas dalam bentuk rangkaian terpadu (IC). Perangkat ini sering digunakan sebagai penguat sinyal-sinyal, baik yang linier maupun non linier terutama dalam sistem—sistem pengaturan dan pengendalian, instrumentasi, komputasi analog. Keuntungan dari pemakaian penguat operasional ini adalah karakteristiknya yang mendekati ideal sehingga dalam merancang rangkaian yang menggunakan penguat ini lebih mudah dan juga karena penguat ini bekerja pada tingkatan yang cukup dekat dengan karakteristik kerjanya.

Karakteristik utama sebuah penguat operasional yang ideal adalah :

- 1) Impedansi masukan yang tak terhingga
- 2) Impedansi keluaran sama dengan nol
- 3) Penguatan Loop terbuka tak terhingga

Op-Amp 358 adalah Operasional Amplifier yang memiliki penguat operasional ganda dan memiliki pengarus daya rendah, modulus umum rentang tegangan input memperluas GND/ VEE, dan satu daya tunggal atau operasi pasokan split. Seri LM358 adalah setara dengan satu setengah dari LM324. Amplifier ini memiliki beberapa keuntungan yang berbeda atas standar jenis penguat operasional dalam aplikasi satu daya tunggal. Op-Amp 358 beroperasi pada tegangan suplai serendah 3,0 V atau setinggi 32 V, dengan arus diam sekitar seperlima dari mereka terkait dengan MC1741 (pada per penguat basis). Berbagai masukan modulus umum termasuk pasokan negatif, sehingga menghilangkan kebutuhan untuk komponen biasing eksternal dalam banyak aplikasi. Tegangan output kisaran juga mencakup tegangan satu daya negatif.

### 2.5.1 Konfigurasi Pin Op-Amp 358



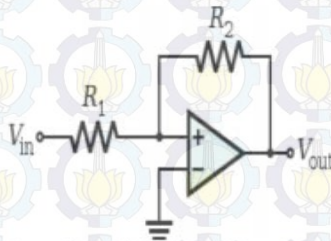
Gamabr 2.18 Konfigurasi Pin Op-Amp 358



- Pin1* yaitu *outputA* yang berfungsi sebagai keluaran dari masukan *Pin1* dan *Pin2*
- Pin2* dan *Pin3* Sebagai *inputA* untuk keluaran *Pin1*
- Pin 4* berfungsi sebagai ground
- Pin5* dan *Pin6* sebagai *inputB* untuk keluaran dari *pin8*
- Pin7* berfungsi sebagai *outputB*
- Pin8* sebagai catu daya positif

### 2.5.2 Non Inverting

Penguat Tak-Membalik (Non-Inverting Amplifier) merupakan penguat sinyal dengan karakteristik dasar sinyal output yang dikuatkan memiliki fasa yang sama dengan sinyal input. Penguat tak-membalik (non-inverting amplifier) dapat dibangun menggunakan penguat operasional, karena penguat operasional memang didesain untuk penguat sinyal baik membalik ataupun tak membalik. Rangkaian penguat tak-membalik ini dapat digunakan untuk memperkuat isyarat AC maupun DC dengan keluaran yang tetap sefase dengan sinyal inputnya



**Gambar 2.19** Rangkaian Non Inverting

Rumus untuk penguatan *non inverting* adalah :

$$A = \frac{V_o}{V_i} = \left( \frac{R_f}{R_i} + 1 \right)$$

A = Penguatan

$V_o$  Tegangan *Output*

$V_i$  = Tegangan *Input*

Rf = Tahanan *Refrensi*

Ri = Tahanan *Input*

## 2.6 Filter RC

Low Pass Filter (LPF) atau Filter Lolos Bawah adalah filter yang hanya melewatkan sinyal dengan frekuensi yang lebih rendah dari frekuensi cut-off ( $f_c$ ) dan akan melemahkan sinyal dengan frekuensi yang lebih tinggi dari frekuensi cut-off ( $f_c$ ). Pada filter LPF yang ideal sinyal dengan frekuensi diatas frekuensi cut-off ( $f_c$ ) tidak akan dilewatkan sama sekali artinya tegangan output adalah 0 volt. Rangkaian low pass filter RC merupakan jenis filter pasif, dengan respon frekuensi yang ditentukan oleh konfigurasi R dan C yang digunakan. Rangkaian dasar LPF dan grafik respon frekuensi LPF sebagai berikut.

$$f_c = \frac{1}{2\pi RC}$$

$f_c$  = Frekuensi cut-off

R = Resistor

C = Capacitor

## 2.7 RSSI ( *Received Signal Strength Indicator* ) [ 7 ]

RSSI adalah Radio-Frequency (RF) yang digunakan sebagai indikator jangkauan penerima kekuatan sinyal, serta parameter yang menunjukan kekuatan dari seluruh sinyal pada *band frequency channel* yang diukur.



**Gambar 2.20** Rangkain RSSI pada *boscam* 5.8 GHz 200mW



### BAB III

## PEMBUATAN DAN PERANCANGAN ALAT

Pada alat Kamera Pemantau pada Pesawat Aeromodelling ini terdapat beberapa langkah dan perangkat utama yang menjadi acuan dalam langkah kerja pembuatan dan perancangan alat. Pertama adalah Pesawat *Aeromodelling* sebagai media terbang kamera, kedua *Remote control* dan *receiver* sebagai kendali dari pesawat, ketiga kamera dan *transmitter* dan *receiver* yang akan ditampilkan di monitor. Selanjutnya bentuk rangkaian kamera yang akan digunakan sebagai alat demonstrasi yaitu pemasangan kamera pada pesawat. Untuk perangkat keras meliputi :

1. Pesawat *Aeromodelling FPV Raptor* 1600mm
2. Rangkaian *transmitter* dan *receiver* untuk kendali pesawat
3. Rangkaian *transmitter* dan *receiver* untuk kamera ke monitor
4. Rangkaian pergerakan kamera
5. Rangkaian indikator sinyal *receiver* kamera

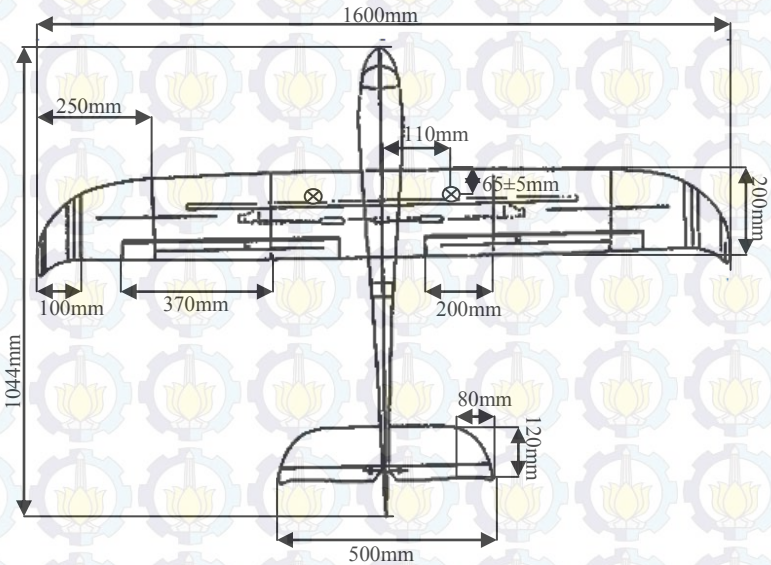
Sedangkan untuk perancangan software program pembacaan data dengan mikrokontroler yang dibuat dan disimulasikan dengan menggunakan *CodeVision AVR* perangkat lunak (*software*) yang akan di-download-kan ke mikrokontroler yang berfungsi untuk pergerakan kamera dan indikator sinyal *receiver* video.

### 3.1 Pesawat *Aeromodelling FPV Raptor* 1600mm

Pesawat ini membutuhkan pergerakan pada sayap *aileron*, *rudder* dan *elevator* yang menggunakan *servo*. Untuk *throttle* menggunakan motor *brushless*. Keseimbangan pesawat saat lepas landas yaitu dengan menentukan *Center of Gravity* (C.G). Idealnya posisi C.G pesawat *FPV Raptor* adalah  $65 \pm 5$ mm dari depan dan 110mm dari tengah sayap pesawat. Untuk membuat C.G dapat ditamabht beban pada *fuselage* atau memindahkan posisi baterai.

*FPV Raptor* memiliki bentang sayap 1600mm, lebar 200mm dan panjang pesawat 1044mm. Untuk ekor sayap horizontal panjang 500mm lebar 120mm dan tinggi sayap vertikal 210mm lebar 200mm. Pesawat ini dapat terbang dengan berat sampai 950gram. Motor yang digunakan adalah *brushless* ukuran diameter 28mm dengan panjang badan 12mm dan motor ini memiliki kecepatan putaran sampai 1400KV permenitnya. Untuk pergerakan sayap menggunakan motor *servo* 9g.

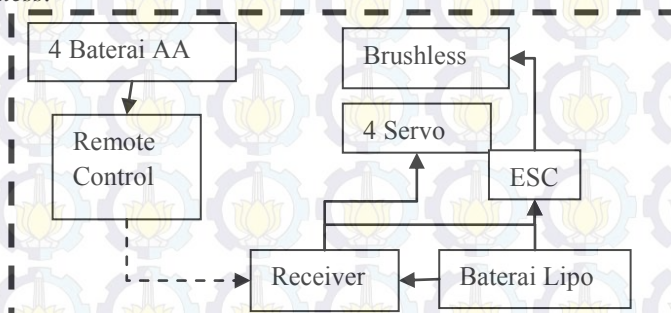




**Gambar 3.1** Ukuran Pesawat FPV Raptor 1600mm

### 3.2 Rangkaian *Transmitter* dan *Receiver* Untuk Kendali Pesawat

Rangkaian ini menghubungkan *remote control* sebagai kendali pesawat dan *receiver* sebagai pengendalian gerakan pesawat. Gerakan pesawat saat di udara ditentukan oleh gerakan dari sayap yang di gerakan oleh motor *servo*. Untuk laju pesawat menggunakan motor *brushless*.



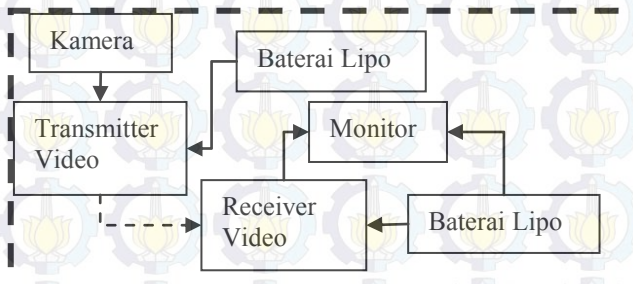
**Gambar 3.2** Diagram Blok sistem *transmitter* dan *receiver* dalam pergerakan pesawat

Untuk dapat mengontrol motor *servo*, *receiver* memberikan pulsa *high* dan pulsa *low* dengan lebar tertentu. Frekuensi yang diperlukan adalah 50 Hz. Pulsa ini dapat dihasilkan dengan *port I/O* pada *receiver*. Untuk mendapat pergerakan *servo* yang diinginkan digunakan metode *Pulse Width Modulation* (PWM). Dengan metode PWM dapat menghasilkan gerakan *servo* yang sesuai dengan keinginan.

Pada motor *Brushless* prinsipnya hampir sama yaitu menggunakan PWM yang mengatur ESC (*Electronic Speed Control*). Semakin lebar pulsa semakin cepat putaran motor.

### 3.3 Rangkaian *Transmitter* dan *Receiver* Kamera ke Monitor

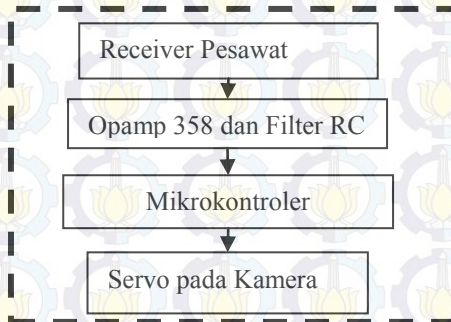
Rangkaian ini menghubungkan kamera ke monitor melalui *transmitter* dan *receiver* yang dikhususkan untuk video. *Transmitter* akan mengirimkan hasil video ke *receiver* lalu ditampilkan ke monitor.



**Gambar 3.3** Diagram Blok sistem *transmitter* dan *receiver* untuk penampilan kamera

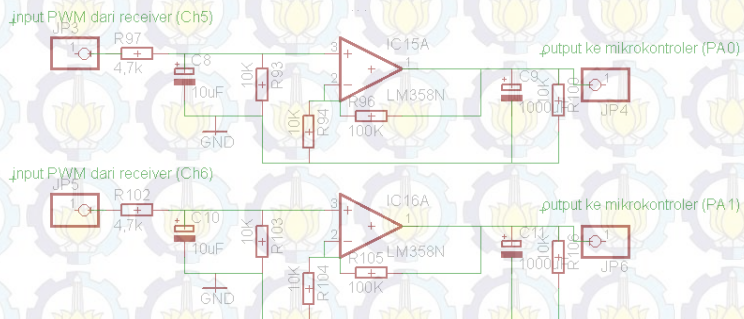
### 3.4 Rangkaian Pergerakan Kamera

Rangkaian pergerakan kamera digunakan untuk mengatur arah kamera sesuai dengan yang diinginkan. Pergerakan kamera berawal dari posisi 0 derajat, kiri, kanan, bawah dan atas. Berikut adalah Diagram blok rangkaian pergerakan kamera.



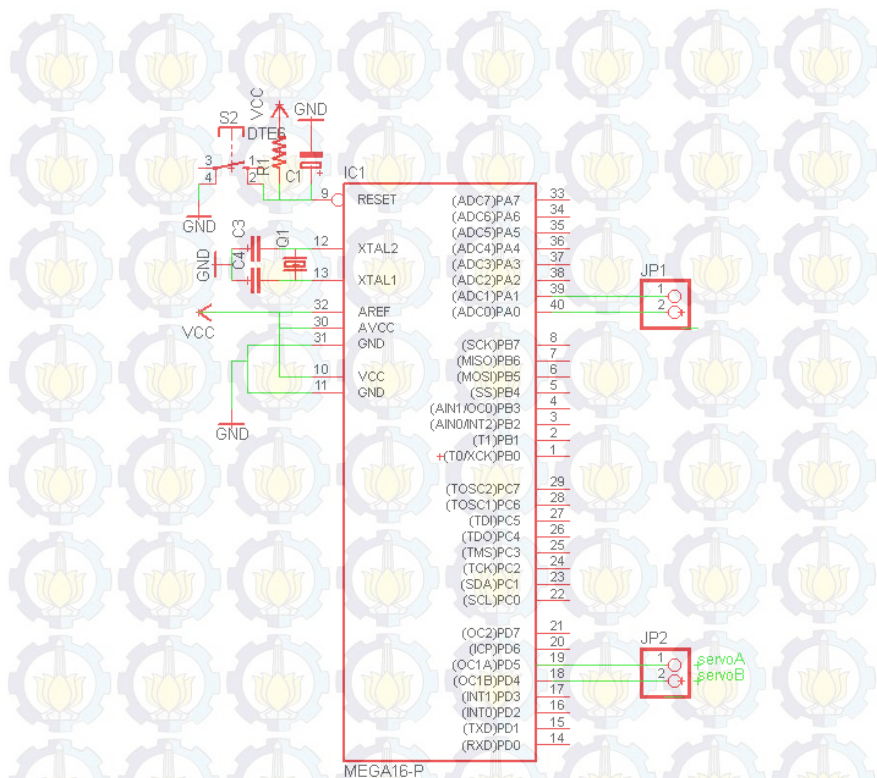
**Gambar 3.4** Diagram Blok Sistem kerja pergerakan kamera

*Receiver remote* pesawat menghasilkan sinyal PWM, rangkaian *OpAmp* 358 yang ditambah rangkaian *filter* RC digunakan untuk menguatkan dan menkonversi sinyal PWM pada *receiver remote* menjadi besaran analog. Minimum sistem dari mikrokontroler AVR ATmega16 ini digunakan sebagai pengontrol dari pergerakan kamera yang berasal dari output *OpAmp* 358 yang diolah menjadi *input* ADC di Mikrokontroler. *Output* Mikrokontroler adalah fungsi *Timer* yang digunakan untuk pergerakan kamera. *Timer* menghasilkan sinyal PWM yang mengatur arah servo. Berikut adalah rangkaian *OpAmp* dan mikrokontroler.



**Gambar 3.5** Rangkaian *OpAmp* 358 untuk pergerakan kamera





**Gambar 3.6** Rangkaian Minimum Sistem AVR ATmega16 pergerakan kamera

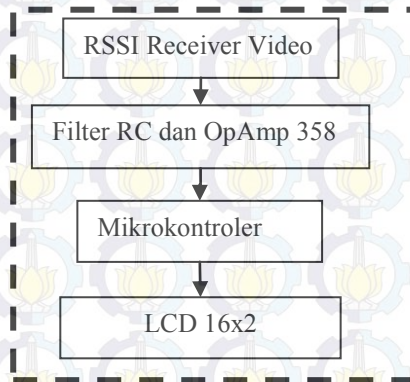
Pada rangkaian mikrokontroler ATmega16 penggunaan masing – masing *port I/O* dalam sistem ini adalah sebagai berikut:

- Port A = masukan ADC
- Port B = -
- Port C = -
- Port D = *Timer* menghasilkan sinyal PWM

### 3.5 Perancangan Rangkaian Indikator Sinyal *Receiver* Video

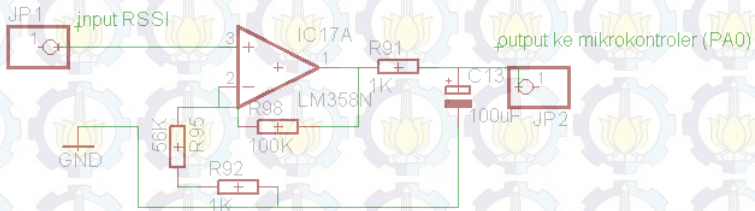
Di dalam *receiver* video, terdapat RSSI (Received Signal Strength Indicator) merupakan indikator kekuatan sinyal. RSSI menghasilkan tegangan yang bervariasi sesuai dengan jarak. Semakin kecil jarak semakin besar tegangan. Karena tegangan RSSI kecil, maka dikuatkan menggunakan *OpAmp* agar skala tegangan yang masuk ke mikrokontroler semakin besar. Tegangan yang dikuatkan masuk ke

mikrokontroler pada *Port A* (ADC) kemudian ditampilkan ke *LCD 16x2* untuk mengetahui seberapa besar kuat sinyal. Berikut adalah Diagram blok indikator sinyal.



**Gambar 3.7** Diagram blok indikator sinyal *receiver video*

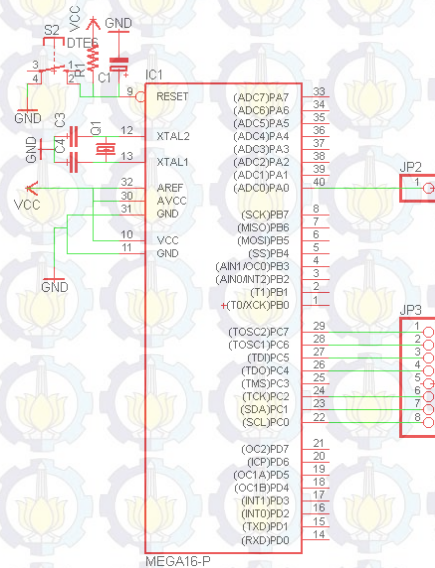
Untuk *OpAmp* ditambahkan rangkaian *filter RC* yang berguna menstabilkan tegangan pada *RSSI receiver video* agar mudah terbaca di *LCD 16x2*. Berikut adalah rangkaian *OpAmp*.



**Gambar 3.8** Rangkaian *OpAmp* indikator sinyal

Setelah terjadi penguatan di *OpAmp*, maka output penguatan masuk ke mikrokontroler. Rangkaian mikrokontroler dapat dilihat pada gambar 3.8. Penggunaan masing – masing *port I/O* Mikrokontroler ATmega16 dalam sistem ini adalah sebagai berikut:

- Port A = masukan ADC
- Port B = -
- Port C = Tampilan LCD
- Port D = -



**Gambar 3.9** Rangkaian Minimum Sistem AVR ATmega16 Indikator Sinyal

### 3.6 Perancangan Software Pergerakan Kamera

Dalam pembuatan *software* untuk diaplikasikan ke dalam sistem pergerakan kamera oleh mikrokontroler menggunakan *CodeVision AVR*, pergerakan digunakan untuk memantau lebih luas suatu wilayah. Berikut ini adalah program dari sistem pergerakan kamera.

```
#include <mega16.h> //Mikrokontroler yang digunakan ATmega16
#include <stdint.h> // interrupt
#include <delay.h> // untuk memberi jeda waktu
#define ADC_VREF_TYPE 0x00
int adc, adc1, i, j, h; // variabel dengan tipe data integer
// Read the AD conversion result
unsigned int read_adc(unsigned char adc_input)
{
    ADMUX=adc_input | (ADC_VREF_TYPE & 0xff);
    // Delay needed for the stabilization of the ADC input voltage
    delay_us(10);
    // Start the AD conversion
    ADCSRA|=0x40;
```



```

// Wait for the AD conversion to complete
while ((ADCSRA & 0x10)==0);
ADCSRA|=0x10;
return ADCW;
}

void servoA(int degree) // prosedur untuk pergerakan servoA
{
int x = 1125 - (int)degree*375.0/90.0; //rumus untuk pergerakan ServoA
OCR1A = x; // Timer1 pada PortD5 menghasilkan PWM
}

void servoB(int degree2) // prosedur untuk pergerakan servoB
{
int y = 1125+(int)degree2*375.0/90.0; //rumus untuk pergerakan ServoB
OCR1B = y; // Timer1 pada PortD4 menghasilkan PWM
}

// Declare your global variables here
void main(void)
{
// Declare your local variables here

// Input/Output Ports initialization
// Port A initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In
Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T
State0=T
PORTA=0x00;
DDRA=0x00; // PortA semua sebagai input

// Port D initialization
// Func7=In Func6=In Func5=Out Func4=Out Func3=In Func2=In
Func1=In Func0=In
// State7=T State6=T State5=0 State4=0 State3=T State2=T State1=T
State0=T
PORTD=0x00;
DDRD=0x30; // PortD4 dan PortD5 sebagai Output

// Timer/Counter 1 initialization
// Clock source: System Clock

```

```

// Clock value: 1500,000 kHz
// Mode: Ph. correct PWM top=ICR1
// OC1A output: Non-Inv.
// OC1B output: Non-Inv.
// Noise Canceler: Off
// Input Capture on Falling Edge
// Timer1 Overflow Interrupt: Off
// Input Capture Interrupt: Off
// Compare A Match Interrupt: Off
// Compare B Match Interrupt: Off
TCCR1A=0xA2;
TCCR1B=0x12;
TCNT1H=0x00;
TCNT1L=0x00;
ICR1H=0x3A;
ICR1L=0x98;
OCR1AH=0x00;
OCR1AL=0x00;
OCR1BH=0x00;
OCR1BL=0x00;

// ADC initialization
// ADC Clock frequency: 750,000 kHz
// ADC Voltage Reference: AREF pin
// ADC Auto Trigger Source: ADC Stopped
ADMUX=ADC_VREF_TYPE & 0xFF;
ADCSRA=0x84;

```

```

while (1) // perulangan program

```

```

{
  adc=read_adc(0); // membaca ADC pada PortA.0
  adc1=read_adc(1); // membaca ADC pada PortA.1

```

```

  if (adc>512 && adc1>512) // Ch5 dan Ch6 On
  {servoA(0); // servoA bergerak 0 derajat
    servoB(3);} // servoB bergerak 3 derajat atau ke atas

```

```

  if (adc<512 && adc1<512) // Ch5 dan Ch6 Off

```

```

{servoA(0); // servoA bergerak 0 derajat
servoB(-40);} // servoB bergerak -40 derajat atau ke bawah

if (adc<512&&adc1>512) // Ch5 Off dan Ch6 On
{
    {for (i=0;i<=90;i++) //perulangan untuk servoA 0 - 90derajat
    {servoA(i); // servoA bergerak ke kanan menuju 90derajat
    delay_ms(30);} // jeda waktu pergerakan kamera setiap 1 derajat
    delay_ms(500); // jeda waktu saat kamera 90 derajat
    }

    for (h=90;h>=0;h--) //perulangan untuk servoA 90-0derajat
    {servoA(h); // servoA bergerak ke kiri menuju 0derajat
    delay_ms(30);} //jeda waktu pergerakan kamera setiap 1 derajat
    delay_ms(500); // jeda waktu saat kamera 0 derajat
    }

if (adc>512&&adc1<512) // Ch5 On dan Ch6 Off
{
    {for (j=0;j>=-90;j--) //perulangan servoA 0 sampai -90derajat
    {servoA(j); // servoA bergerak ke kiri menuju -90derajat
    delay_ms(30);} //jeda waktu pergerakan kamera setiap 1 derajat
    delay_ms(500); // jeda waktu saat kamera -90 derajat
    }

    for (h=-90;h<=0;h++)//perulangan servoA -90 sampai 0derajat
    {servoA(h); // servoA bergerak ke kanan menuju 0derajat
    delay_ms(30);} //jeda waktu pergerakan kamera setiap 1 derajat
    delay_ms(500); // jeda waktu saat kamera 0derajat
    }
}
}

```

Dari program di atas dapat dijabarkan rinciannya sebagai berikut :

1. Ketika alat dihidupkan, maka hal pertama adalah mengenali *port* mana saja dari mikrokontroler yang digunakan dan masing-masing fungsi dari *port* tersebut. Kemudian mengetahui fitur apa saja dari mikrokontroler yang digunakan di dalamnya. Fitur yang digunakan dalam pergerakan kamera adalah ADC dan Timer1.



2. Tahap berikutnya yaitu terdapat rumus yang didapat dari pengaturan register timer pada mikrokontroler agar dapat menghasilkan pulsa dengan lebar yang diinginkan. Dua parameter utama yang diperlukan untuk mencari konstanta-konstanta timer adalah nilai clock mikrokontroler dan nilai clock timer yang ditentukan dari *prescaler*. Lalu Pulsa dihasilkan dari pin OCR pada mikrokontroler.

Cara menentukan rumus tersebut adalah menentukan *clock* yang dipakai mikrokontroler. Dalam rangkaian mikrokontroler menggunakan *clock* eksternal 12 MHz. Mode PWM yang digunakan adalah *phase correct PWM top = ICR1* dan nilai *clock* 1500 KHz. Dengan demikian kenaikan *counter* pada *timer* dapat dihitung dari periodenya yaitu  $1/1500 \text{ KHz} = 6,67 \times 10^{-7} \text{ s}$ . Periode PWM servo adalah 20ms karena menggunakan mode *Phase Correct PWM* maka periodenya dua kali kenaikan counter timer, sehingga nilai counter 10ms. Nilai top dari timer bergantung dari ICR1 dimana nilainya dihitung dari pembagian total waktu yang kita inginkan dengan periode timer yaitu  $10\text{ms} / 6,67 \times 10^{-7} \text{ s} = 15000$ . Nilai ini perlu dimasukkan ke register ICR1 dalam bilangan hexadesimal yaitu 0x3A98. Berikut pengaturan timer1.

Chip	Ports	External IRQ	Timers
Timer0	Timer1	Timer2	Watchdog
Clock Source: System Clock			
Clock Value: 1500,000 kHz			
Mode: Ph. correct PWM top=ICR1			
Out. A: Non-Inv. Out. B: Non-Inv.			
Input Capt.: <input type="checkbox"/> Noise Cancel			
Interrupt on: <input type="checkbox"/> Timer1 Overflow			
Value: 0 h Inp. Capture: 3a98 h			
Comp. A: 0 h B: 0 h			

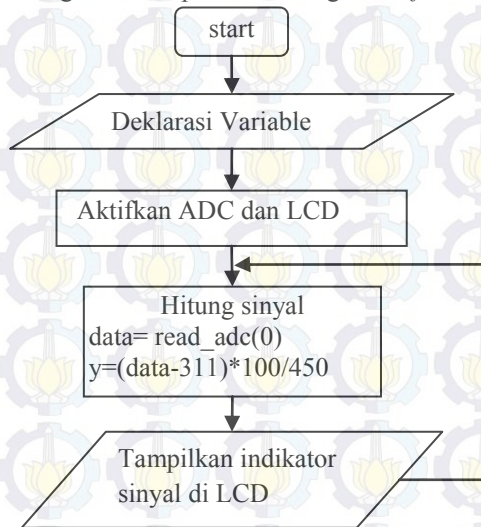
**Gambar 3.10** Pengaturan timer untuk pergerakan servo

Untuk mengatur sudut servo  $-90^\circ$  diperlukan pulsa on 1ms. Karena mode *phase correct* PWM maka menjadi 0,5ms. Sehingga  $0,5\text{ms} / 6,67 \times 10^{-7} \text{ s} = 750,7$  dibulatkan 750. Untuk mengatur sudut servo  $90^\circ$  diperlukan pulsa on 2ms. Karena mode *phase correct* PWM maka menjadi 1ms. Sehingga  $1\text{ms} / 6,67 \times 10^{-7} \text{ s} = 1500$ . Pada flowchart angka 1125 adalah saat servo  $0^\circ$ .

3. Pada tahap perulangan terus menerus (*while*) terdapat tahap pembacaan ADC dan pergeseran pergerakan derajat servo. Pembacaan ADC disesuaikan dengan keluaran dari OpAmp. Jika ADC kurang dari 512 maka servo  $0^\circ$ . Jika ADC lebih dari 512 maka servo  $-90^\circ$ . Untuk ADC1, jika ADC1 lebih dari 512 maka servo  $90^\circ$

### 3.7 Perancangan Software Sinyal Indikator Video

Dalam pembuatan *software* untuk diaplikasikan ke dalam indikator sinyal menggunakan *CodeVision AVR*. Berikut ini adalah algoritma dari indikator sinyal. Program ditampilkan dalam gambar *flowchart* berikut.



**Gambar 3.11** Algoritma indikator sinyal

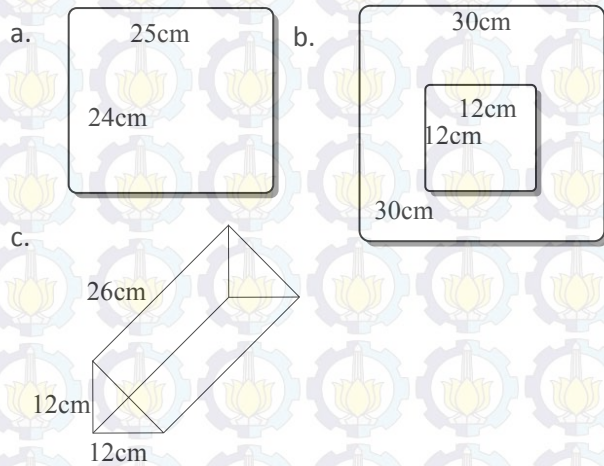
Dari gambar algoritma *Flowchart* di atas dapat dijabarkan rinciannya sebagai berikut :

1. Ketika alat dihidupkan, maka hal pertama adalah mengenali *port* mana saja dari mikrokontroler yang digunakan dan masing-masing fungsi dari *port* tersebut. Kemudian mengetahui fitur apa saja dari mikrokontroler yang digunakan di dalamnya. Fitur yang digunakan dalam pergerakan kamera adalah ADC dan LCD.
2. Pada tahap perulangan terus menerus (*while*) terdapat tahap pembacaan ADC. Pembacaan ADC disesuaikan dengan keluaran dari *OpAmp*. Rumus sinyal didapat dari tegangan yang keluar dari *OpAmp* yaitu saat sinyal lemah 1,55v dan saat sinyal kuat 3,7v. Untuk nilai ADC dari 1,55v adalah 317,13 atau 317 dan nilai ADC dari 3,7 adalah 757,02 atau 757. Rumus dalam flowchart menentukan berapa persen sinyal yang didapat. Saat sinyal lemah mendekati 0% dan saat sinyal kuat mendekati 100%.
3. Tampilkan indikator di LCD.

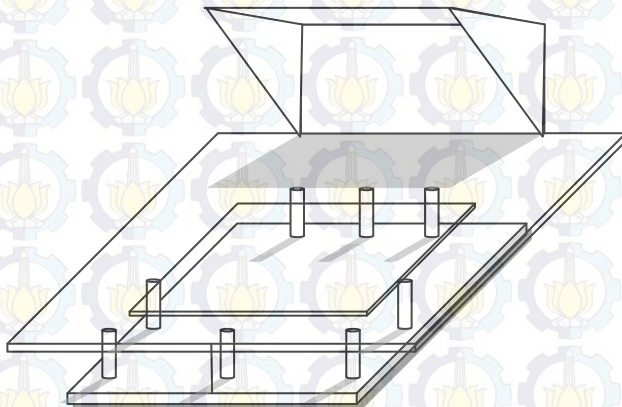
### **3.8 Perancangan penyangga remote control dan monitor**

Dalam pembuatan *Tray remote* terdiri dari dua bagian kotak dan setengah kubus untuk melindungi monitor dari cahaya matahari. Kotak yang pertama berfungsi sebagai dasar dari penempatan *remote control* berukuran 25cm x 24cm. Kotak yang kedua menempel pada kotak yang pertama dengan ukuran 30cm x 30cm dan berlubang ditengahnya dengan ukuran 12cm x 12 cm berfungsi penyangga monitor. untuk melindungi monitor dari cahaya digunakan setengah kubus berwarna hitam berukuran 26cm x 12cm x 12cm . *remote tray* ini dibuat menggunakan akrilik, penyangganya menggunakan *spacer* dan baut. Berikut gambar penyangga *remote control* dan monitor.





**Gambar 3.12** a. Kotak pertama terletak di bawah, b. kotak kedua terletak di atas dan c.udukan monitor terletak di atas kotak kedua



**Gambar 3.13** remote tray

Gambar diatas adalah kotak dari gambar 3.13 yang sudah dirancang. Kedua kotak dihubungkan paralel menggunakan spacer. Remote diletakkan di tengah pada kotak yang sudah dilubangi. Monitor

diletakkan pada setengah balok yang berfungsi melindungi dari cahaya matahari.

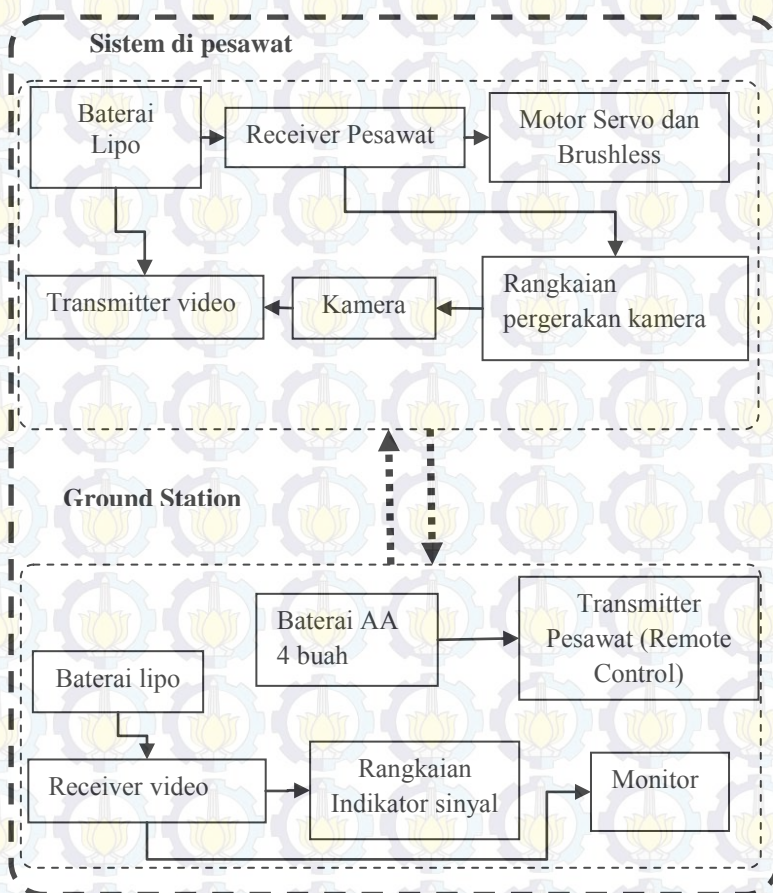
### 3.9 Sistem Secara Keseluruhan

Dari sistem secara keseluruhan dibagi menjadi dua yaitu sistem *ground station* dan sistem di pesawat. Pada *ground station* terdapat *receiver* video, *transmitter* pesawat, monitor dan indikator sinyal. Pada pesawat terdapat *transmitter* video, *receiver* pesawat, kamera dan rangkaian pergerakan kamera.

Pada sistem di pesawat sumber daya menggunakan baterai Lipo yang berguna untuk menyalakan *receiver* pesawat, motor sebagai pergerakan pesawat, *transmitter* video dan rangkaian pergerak kamera. *Receiver* pesawat berguna untuk mengatur motor yang digunakan untuk pergerakan kamera dan pesawat. Fungsi kamera dan *transmitter* video adalah menangkap gambar yang nantinya di tampilkan di monitor.

Pada *ground station* baterai Lipo digunakan sebagai *supply* untuk *receiver* video, rangkaian indikator sinyal dan monitor. Sedangkan untuk baterai AA digunakan untuk *supply remote control* sebagai pengendali pesawat dan pergerakan kamera. *Receiver* video berguna menangkap hasil kamera yang dipancarkan *transmitter* video dan nantinya ditampilkan di monitor. Pada indikator sinyal digunakan untuk mengetahui seberapa kuat *receiver* menangkap hasil gambar dari kamera

Berikut Diagram blok sistem secara keseluruhan.



**Gambar 3.14** Diagram blok sistem secara keseluruhan



## BAB IV

### PENGUJIAN DAN ANALISA DATA

Untuk mengetahui apakah tujuan dari pembuatan alat ini bekerja atau tidak, perlu dilakukan pengujian dan analisa alat. Pengujian tiap bagian penyusun sistem kamera pemantau meliputi sumber tegangan, sinyal yang dapat diterima oleh *receiver* video, *receiver remote* untuk pergerakan kamera, *OpAmp* 358 sebagai penguat juga sebagai input mikrokontroler dan mikrokontroler sendiri.

#### 4.1 Pengujian Sumber Tegangan

Pengujian *supply* di *ground station* merupakan sumber daya dari baterai Lipo 11,3 V yang membutuhkan *regulator* 7809 untuk mendapatkan spesifikasi tegangan 9 Vdc yang digunakan untuk *receiver* video. Pada rangkaian sismin, dan *OpAmp* membutuhkan tegangan 5 Vdc, sehingga dibutuhkan *regulator* 7805 untuk menjadikan tegangan 5 Vdc. Sebelum sumber tegangan dari baterai Lipo 11,3 V masuk ke *regulator* 7805, tegangan masuk ke *regulator* 7809 diturunkan menjadi 9 Vdc lalu masuk *regulator* 7805 menjadi 5 Vdc. Pengujian dilakukan dengan menggunakan *AVO*meter. Untuk mengetahui apakah tegangan sesuai dengan yang diinginkan, maka perlu adanya pengujian yaitu dengan hasil dalam tabel sebagai berikut.

NO	Rangkaian	Tegangan yang diinginkan	Tegangan Terukur	Error (V)	%Error
1	7805	5 V	5,09 V	0,09	1,8 %
2	7809	9 V	9.32 V	0,32	3,55 %

**Tabel 4.1** Pengujian sumber tegangan.

Untuk *supply* pada sistem pesawat menggunakan baterai Lipo 11,3 V untuk menggerakan motor *Brushless*, *receiver* pesawat, sismin dan *Opamp* ditambah baterai Lipo 7,4 V untuk *transmitter* video. Pada baterai Lipo 11,3 V diturunkan tegangannya menggunakan rangkaian yang ada di ESC (Electronic Speed Control) menjadi 5,2 V yang digunakan untuk *receiver* pesawat, sismin dan *OpAmp*.

## 4.2 Pengujian Sinyal Transmitter dan Receiver Video

Di dalam *receiver* video terdapat RSSI (*Receiver Strenght Signal Indicator*). RSSI adalah indikator seberapa besar sinyal dapat ditangkap. RSSI menghasilkan tegangan yang bervariasi sesuai dengan jarak. Semakin kecil jarak antara *receiver* dan *transmitter* maka semakin besar tegangan dan semakin besar jarak *receiver* dan *transmitter* maka semakin kecil tegangan.

Selanjutnya pengujian *Opamp* 358 untuk penguatan tegangan pada RSSI. tegangan RSSI diberi penguatan *Opamp* sebesar 3 kali. penguatan digunakan agar skala pembacaan pada ADC mikrokontroler lebih lebar.

pengujian berikutnya dilakukan dengan mengukur seberapa kuat *transmitter* dan *receiver* dapat mengirimkan gambar ke monitor. Pengujian dilakukan dengan dua cara yaitu saat *receiver* dan *transmitter* video terhalang dan saat tidak ada halangan. Pengujian saat terhalang adalah dengan menghalangi *receiver* dan *transmitter* video dengan dinding atau bangunan. Pengujian sinyal saat terhalang didapatkan hasil dalam tabel berikut:

NO.	Jarak (m)	RSSI (V)	OpAmp 358(V)	Sinyal (%)
1	1	1,35	3,73	98
2	3	1,05	3,07	68
3	5	0,85	2,36	36
4	10	0,53	1,63	3

**Tabel 4.2** Pengujian *transmitter* dan *receiver* video saat terhalang

Selanjutnya adalah pengujian *transmitter* dan *receiver* video tanpa halangan. Pengujian dilakukan di lapangan yang sebelumnya panjang lapangan diukur menggunakan meteran bangunan dan odometer pada sepeda motor. Hasil pengujian sinyal adalah dalam tabel berikut.

NO.	Jarak (m)	RSSI (V)	OpAmp 358 (V)	Sinyal (%)
1	0	1,37	3,73	99
2	50	1,05	2,90	61
3	100	0,95	2,62	49
4	150	0,86	2,40	37
5	200	0,81	2,20	30
6	250	0,75	2,07	20



7	300	0,73	1,99	18
8	350	0,70	1,90	15
9	400	0,64	1,74	12
10	450	0,64	1,74	12
11	500	0,61	1,67	8

**Tabel 4.3** Pengujian *transmitter* dan *receiver* video di lapangan saat tidak terhalang

Untuk mengetahui berapa persen sinyal monitor dapat menangkap gambar, perlu dilakukan pengujian menghubungkan indikator sinyal video dengan monitor. Indikator sinyal antara 37-100 % gambar masih terlihat jelas. Sinyal antara 36-9% gambar terlihat kabur. Saat sinyal hanya 8 % ke bawah maka monitor sudah mulai kehilangan gambar dan kadang gambar tidak bisa ditampilkan lagi. Pengujian sinyal yang dihubungkan dengan monitor dapat dilihat pada gambar 4.1.



**Gambar 4.1** Tampilan monitor saat sinyal 100% dan 8%

#### 4.3 Pengujian *Receiver Remote Control*

Pengujian dilakukan dengan mengukur tegangan keluaran *receiver remote control* menggunakan *AVOmeter*. Tegangan yang keluar dari *receiver* ini digunakan untuk *supply* sismin dan *OpAmp* untuk pergerakan kamera. Tabel hasil pengukuran tegangan *receiver* adalah sebagai berikut.

Rangkaian	Tegangan yang diinginkan	Tegangan Terukur	%Error
Tegangan <i>receiver</i>	5 V	5,2 V	4 %

**Tabel 4.4** Pengujian Tegangan *Receiver remote control*

Selanjutnya adalah pengujian sinyal *PWM* pada Ch5 dan Ch6 pada *receiver remote control* menggunakan Oskiloskop. Ch5 dan Ch6 berguna untuk pergerakan kamera, dimana saat Ch5 pulsa lebar maka



bergerak ke kiri, saat Ch6 pulsa lebar maka bergerak ke kanan dan saat Ch5 dan Ch6 pulsa sempit maka kamera posisi netral. Gambar 4.2 adalah hasil Oskiloskop sinyal *PWM receiver*.



**Gambar 4.2** *PWM* Ch5 dan Ch6 Off dan On

Pada pengujian jarak *receiver* dilakukan di jalan menggunakan odometer sepeda motor. Kemungkinan jarak dapat lebih jauh lagi saat di udara. Pengujian jarak *receiver* dapat dilihat dalam tabel berikut.

NO.	Jarak (m)	Kondisi
1	100	Hidup
2	200	Hidup
3	300	Hidup
4	400	Hidup
5	500	Mati

**Tabel 4.5** Pengujian Jarak *receiver* pesawat

#### 4.4 Pengujian pada Pergerakan Kamera

*Filter RC* digunakan untuk merubah *PWM* pada *receiver* pesawat menjadi besaran analog. Besaran analog tadi digunakan untuk masukan *OpAmp* yang nantinya dikuatkan. Pengukuran *filter* tadi dilakukan menggunakan *AVOmeter*.

NO	Tegangan dari	Tegangan Terukur
1	Ch5 off	0,17 V
2	Ch5 on	0,39 V
3	Ch6 off	0,12 V
4	Ch6 on	0,26 V

**Tabel 4.6** Pengujian hasil output Filter *RC*

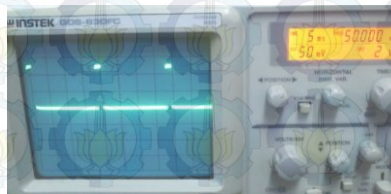
Pengujian dilakukan dengan mengukur tegangan keluaran yang sudah dikuatkan 11 kali dari penguatan OpAmp. Hasil data yang didapat keluaran dengan rangkaian *filter* RC dan OpAmp 358 penguatan 11 kali dalam tabel berikut.

NO	Tegangan dari	Tegangan yang diinginkan	Tegangan Terukur	Error (V)	%Error
1	Ch5 off	1,87 V	1,33 V	0,54	28,87 %
2	Ch5 on	4,29 V	2,87 V	1,42	33,10 %
3	Ch6 off	1,32 V	1,35 V	0,03	2,27 %
4	Ch6 on	2,86 V	2,88V	0,02	0,69 %

**Tabel 4.7** Pengujian setelah penguatan 11 kali pada OpAmp

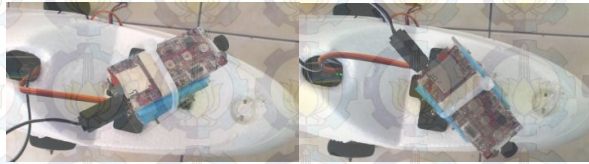
Dari penguatan OpAmp tersebut digunakan untuk fungsi ADC ke mikrokontroler. Tujuan penguatan adalah agar mudah membedakan pembacaan ADC saat kamera bergerak ke kanan, ke kiri, bawah dan atas.

Selanjutnya pengujian hasil keluaran mikrokontroler yaitu sinyal *PWM* yang dihasilkan di PortD 4 dan PortD 5 dimana PortD tadi difungsikan sebagai Timer1. Hasil *PWM* dari mikrokontroler dapat dilihat pada gambar 4.3.



**Gambar 4.3** Hasil *PWM* pada mikrokontroler

Gambar diatas menunjukkan sinyal *PWM* untuk menggerakkan *Servo* pada kamera. Lebar sinyal yang dihasilkan 1ms untuk bergerak ke kiri dan 2ms bergerak ke kanan. Untuk pergerakan kamera dapat dilihat pada gambar 4.4.



**Gambar 4.4** Kamera bergerak ke kiri dan ke kanan

Pergerakan kamera membutuhkan waktu untuk ke kiri dan ke kanan. Pengujian lamanya kecepatan pergerakan ditampilkan dalam tabel berikut.

NO	Pergerakan Kamera	Kecepatan (rad/s)
1	Kiri	15°
2	Kanan	15°

**Tabel 4.8** Pengujian Kecepatan Pergerakan kamera

Pada pergerakan kamera atas dan bawah

#### 4.5 Pengujian Rangkaian Minimum Mikrokontroler

Dalam tahap ini, pengujian dilakukan dengan memberikan program sederhana dimana seluruh *port* pada ATmega16 diberikan logika 1 melalui program yang kemudian di-*download* ke mikrokontroler. Tegangan *supply* untuk rangkaian sistem minimum mikrokontroler adalah sebesar 5 Volt. Pengukuran *output* pada mikrokontroler dilakukan dengan cara mengukur tegangan pada tiap-tiap kaki pada mikrokontroler terhadap *ground*. Kabel merah *AVO*meter dihubungkan dengan kaki mikrokontroler dan kaki warna hitam dihubungkan dengan *ground*. Berikut hasil pengujian rangkaian sistem minimum mikrokontroler.

Port A	Vout (Volt)	Port B	Vout (Volt)	Port C	Vout (Volt)	Port D	Vout (Volt)
0	5,01	0	5,01	0	5,01	0	5,01
1	5,01	1	5,01	1	5,01	1	5,01
2	5,01	2	5,01	2	5,01	2	5,01
3	5,01	3	5,01	3	5,01	3	5,01
4	5,01	4	5,01	4	5,01	4	5,01
5	5,01	5	5,01	5	5,01	5	5,01



6	5,01	6	5,01	6	5,01	6	5,01
7	5,01	7	5,01	7	5,01	7	5,01

**Tabel 4.9** Pengujian Rangkaian Minimum

#### 4.6 Pengujian LCD

Dalam pengujian LCD yaitu dengan cara memberikan program dengan perintah menampilkan karakter pada LCD. Pengujian ini dilakukan bekerja tanpa adanya simbol – simbol tambahan yang muncul karena masalah pada penyambungan LCD ke Mikrokontroler. Berikut adalah Gambar 4.5 hasil dari *running* program menampilkan karakter dan angka.



**Gambar 4.5** Tampilan pada LCD



*Halaman ini sengaja dikosongkan*

## BAB V PENUTUP

Dari pengukuran dan pengujian alat yang telah dilakukan, baik dari perangkat lunak ataupun perangkat keras dapat ditarik kesimpulan dan saran sebagai berikut.

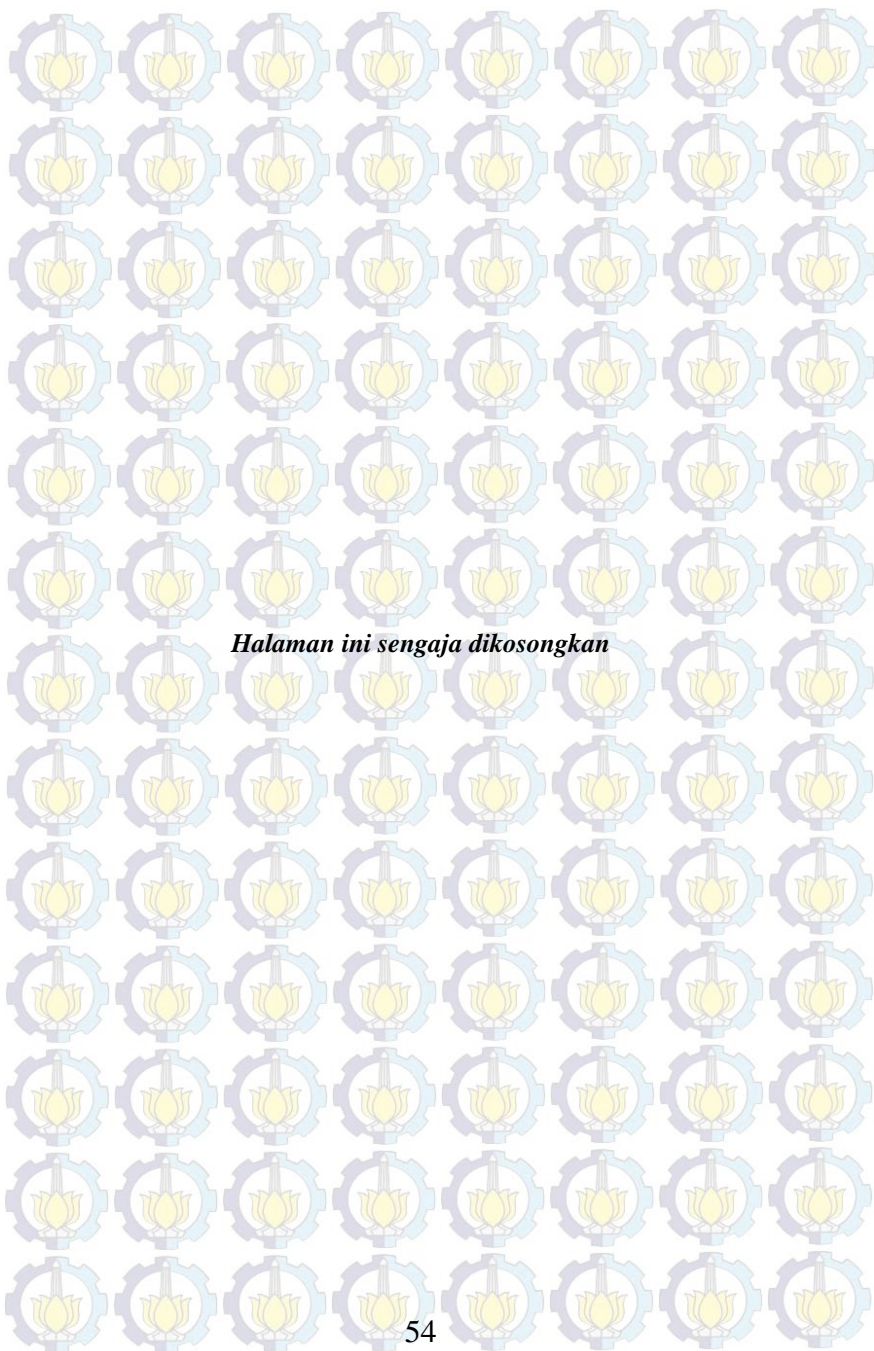
### 5.1 Kesimpulan

1. Sistem untuk pergerakan kamera dapat digunakan sesuai dengan keinginan kontrol.
2. Pemotretan udara secara langsung dapat dilakukan pada ketinggian sejajar dengan gedung.
3. Jarak maksimum yang dapat ditempuh *remote control* terhadap *receiver* untuk pergerakan kamera adalah  $\pm 500\text{m}$ .

### 5.2 Saran

1. Diutamakan dalam pemilihan tempat yang luas agar pesawat dapat mengantisipasi kerusakan dalam pendaratan darurat.
2. Untuk mengukur seberapa jauh pemantauan secara langsung, hendaknya mengetahui hasil jarak pantauan. Untuk memudahkan mengetahui jarak *real* bisa menggunakan GPS.
3. Untuk pengembangan alat ini selanjutnya dapat dikembangkan menjadi sistem autopilot.
4. Menambahkan penguat antena agar jarak jangkauan video lebih jauh lagi.





*Halaman ini sengaja dikosongkan*

## DAFTAR PUSTAKA

1. Fillah, Adithya Firmanda, “Pesawat UAV (Unmanned Aerial Vehicle) Sebagai Alat Pengambilan Gambar dari Udara”. Tugas akhir, Program S1 Desain Produk FTSP-ITS, Surabaya, 2008
2. Craig S, Issod (2014),”Getting Started with Hobby Quadcopters and Drones”. Dari, [https:// play .google. com/store/books/details?id = hcPDAwAAQBAJ&rdid=book-hcPDAwAAQBAJ&rdot =1&source = gbs\\_vpt\\_read&pcampaignid =books\\_booksearch\\_viewport](https://play.google.com/store/books/details?id=hcPDAwAAQBAJ&rdid=book-hcPDAwAAQBAJ&rdot=1&source=gbv_vpt_read&pcampaignid=books_booksearch_viewport).
3. Budiharto, Widodo (2007), “Panduan Praktikum Mikrokontroler AVR ATmega16”, Jakarta: PT Elex Media Komputindo.
4. Wowok (2008), ”Antena Wireless Untuk Rakyat”, Yogyakarta: Penerbit Andi Yogyakarta.
5. Succar, Marcio (2013), “HD Wing Camera in FPV” dari [www.rcnaveia.com.br/2013/10/hd-wing-camera-in-fpv.html](http://www.rcnaveia.com.br/2013/10/hd-wing-camera-in-fpv.html)
6. Saripudin (2011), “Kendali Motor Servo dengan Pulse Width Modulation (PWM) pada Mikrokontroler AVR”. dari <http://pudintekel.blogspot.com/2011/09/kendali-motor-servo-dengan-pulse-width.html>
7. Rcgroups (2010), “IBCrazy’s DIY antenna tracker!!!” dari <http://www.rcgroups.com/forums/showthread.php?t=1337608>



*Halaman Ini Sengaja Dikosongkan*



### ***Listing Program CodeWizardAVR Pergerakan Kamera***

/\*\*\*\*\*

This program was produced by the  
CodeWizardAVR V2.05.3 Standard  
Automatic Program Generator

© Copyright 1998-2011 Pavel Haiduc, HP InfoTech s.r.l.  
<http://www.hpinfotech.com>

Project :  
Version :  
Date : 17/06/2014  
Author : tyery08  
Company : embeeminded.blogspot.com  
Comments:

Chip type : ATmega16  
Program type : Application  
AVR Core Clock frequency: 12,000000 MHz  
Memory model : Small  
External RAM size : 0  
Data Stack size : 256

\*\*\*\*\*/

```
#include <mega16.h>
#include <stdint.h>
#include <delay.h>
#define ADC_VREF_TYPE 0x00
int adc, adc1,i,j,h;
// Read the AD conversion result
unsigned int read_adc(unsigned char adc_input)
{
    ADMUX=adc_input | (ADC_VREF_TYPE & 0xff);
    // Delay needed for the stabilization of the ADC input voltage
    delay_us(10);
    // Start the AD conversion
    ADCSRA|=0x40;
    // Wait for the AD conversion to complete
    while ((ADCSRA & 0x10)==0);
    ADCSRA|=0x10;
```

```

return ADCW;
}
void servoA(int degree)
{
    int x = 1125 - (int)degree*375.0/90.0;
    OCR1A = x;
}
void servoB(int degree2)
{
    int y = 1125 + (int)degree2*375.0/90.0;
    OCR1B = y;
}
// Declare your global variables here
void main(void)
{
    // Declare your local variables here

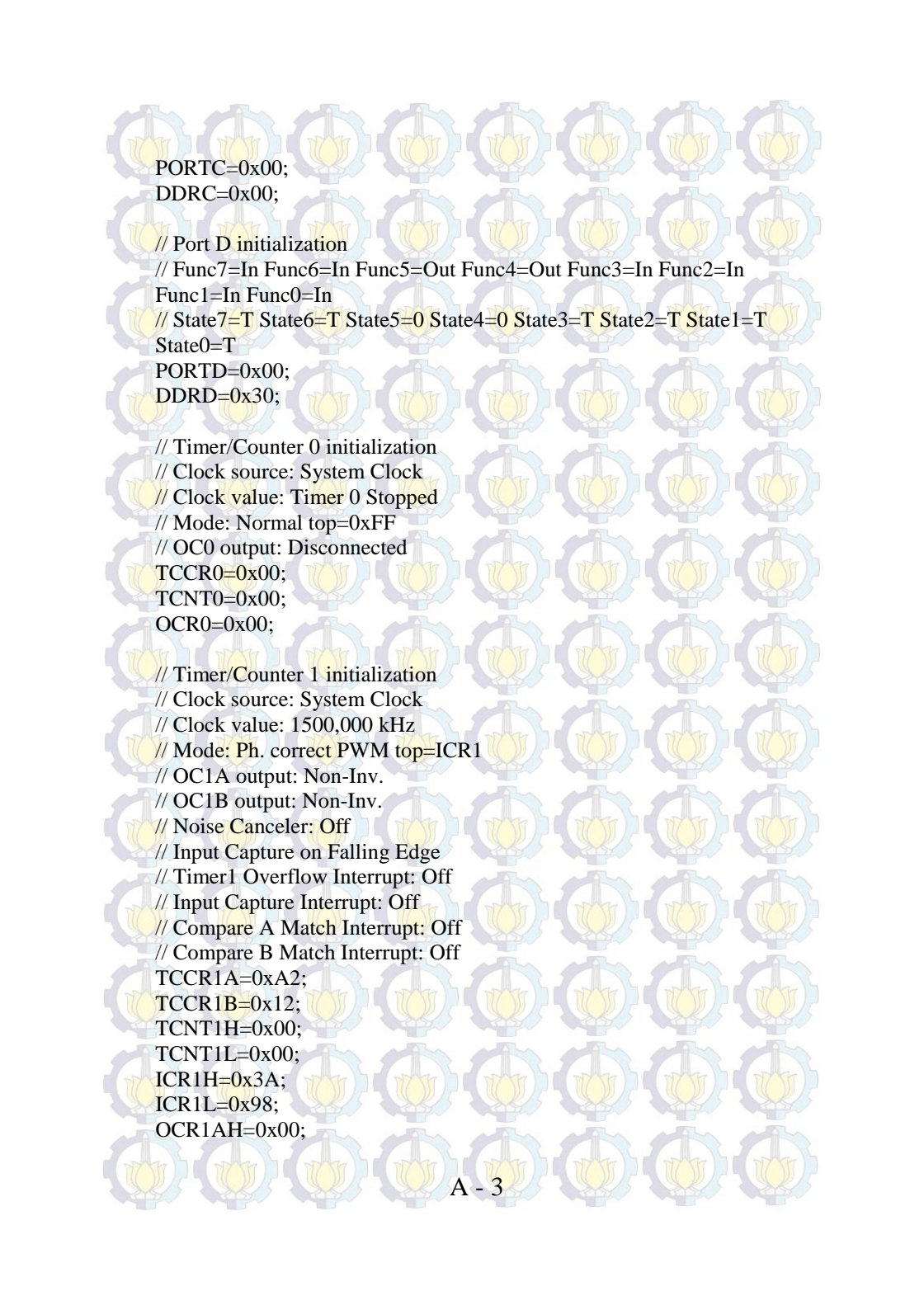
    // Input/Output Ports initialization
    // Port A initialization
    // Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In
    Func1=In Func0=In
    // State7=T State6=T State5=T State4=T State3=T State2=T State1=T
    State0=T
    PORTA=0x00;
    DDRA=0x00;

    // Port B initialization
    // Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In
    Func1=In Func0=In
    // State7=T State6=T State5=T State4=T State3=T State2=T State1=T
    State0=T
    PORTB=0x00;
    DDRB=0x00;

    // Port C initialization
    // Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In
    Func1=In Func0=In
    // State7=T State6=T State5=T State4=T State3=T State2=T State1=T
    State0=T

```





```

PORTC=0x00;
DDRC=0x00;

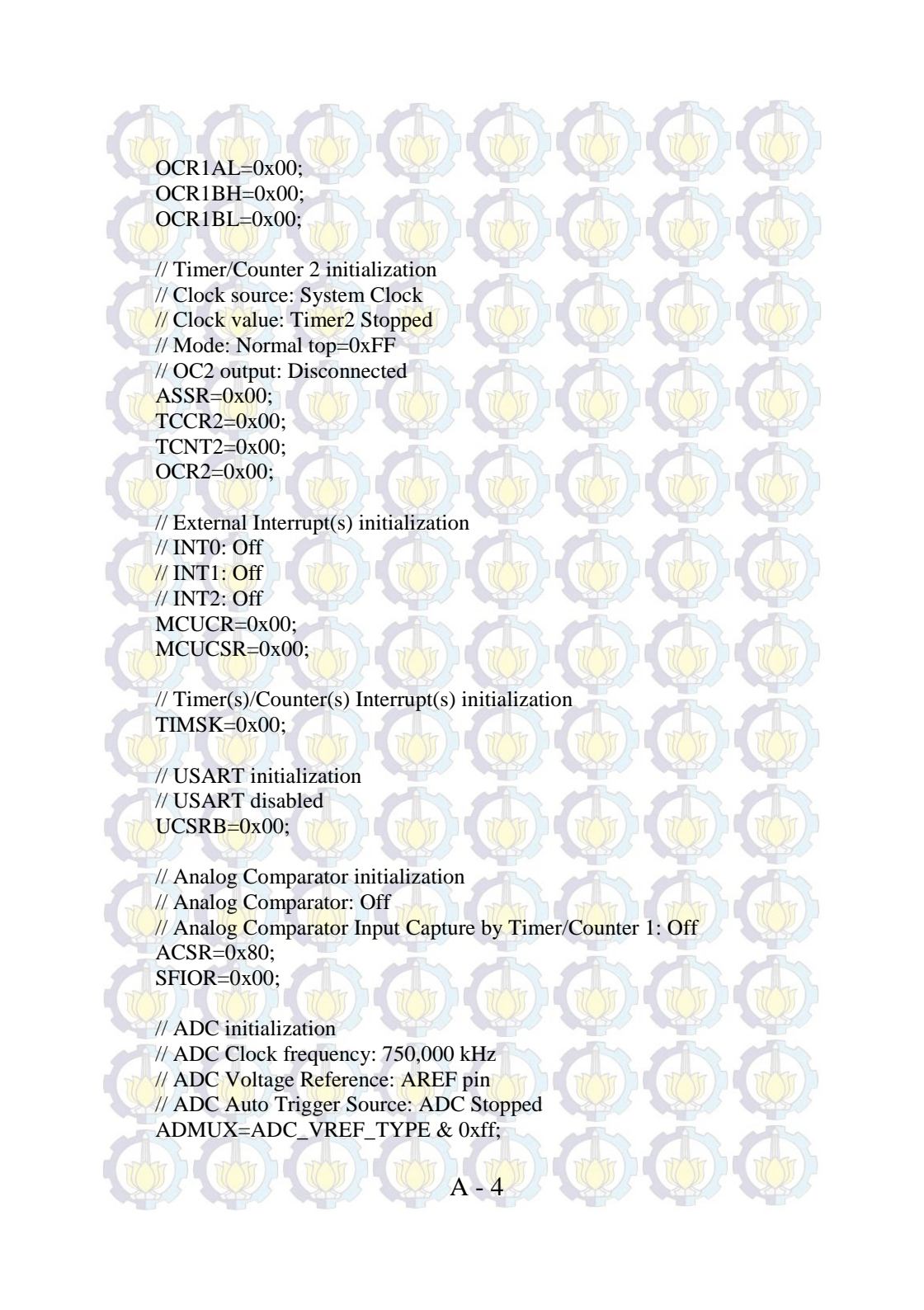
// Port D initialization
// Func7=In Func6=In Func5=Out Func4=Out Func3=In Func2=In
Func1=In Func0=In
// State7=T State6=T State5=0 State4=0 State3=T State2=T State1=T
State0=T
PORTD=0x00;
DDRD=0x30;

// Timer/Counter 0 initialization
// Clock source: System Clock
// Clock value: Timer 0 Stopped
// Mode: Normal top=0xFF
// OC0 output: Disconnected
TCCR0=0x00;
TCNT0=0x00;
OCR0=0x00;

// Timer/Counter 1 initialization
// Clock source: System Clock
// Clock value: 1500,000 kHz
// Mode: Ph. correct PWM top=ICR1
// OC1A output: Non-Inv.
// OC1B output: Non-Inv.
// Noise Canceler: Off
// Input Capture on Falling Edge
// Timer1 Overflow Interrupt: Off
// Input Capture Interrupt: Off
// Compare A Match Interrupt: Off
// Compare B Match Interrupt: Off
TCCR1A=0xA2;
TCCR1B=0x12;
TCNT1H=0x00;
TCNT1L=0x00;
ICR1H=0x3A;
ICR1L=0x98;
OCR1AH=0x00;

```





```
OCR1AL=0x00;  
OCR1BH=0x00;  
OCR1BL=0x00;
```

```
// Timer/Counter 2 initialization  
// Clock source: System Clock  
// Clock value: Timer2 Stopped  
// Mode: Normal top=0xFF
```

```
// OC2 output: Disconnected  
ASSR=0x00;  
TCCR2=0x00;  
TCNT2=0x00;  
OCR2=0x00;
```

```
// External Interrupt(s) initialization
```

```
// INT0: Off  
// INT1: Off  
// INT2: Off  
MCUCR=0x00;  
MCUCSR=0x00;
```

```
// Timer(s)/Counter(s) Interrupt(s) initialization
```

```
TIMSK=0x00;
```

```
// USART initialization
```

```
// USART disabled  
UCSRB=0x00;
```

```
// Analog Comparator initialization
```

```
// Analog Comparator: Off  
// Analog Comparator Input Capture by Timer/Counter 1: Off  
ACSR=0x80;  
SFIOR=0x00;
```

```
// ADC initialization
```

```
// ADC Clock frequency: 750,000 kHz  
// ADC Voltage Reference: AREF pin  
// ADC Auto Trigger Source: ADC Stopped  
ADMUX=ADC_VREF_TYPE & 0xff;
```



```
ADCSRA=0x84;
```

```
// SPI initialization
```

```
// SPI disabled
```

```
SPCR=0x00;
```

```
// TWI initialization
```

```
// TWI disabled
```

```
TWCR=0x00;
```

```
while (1)
```

```
{
```

```
  adc=read_adc(0);
```

```
  adc1=read_adc(1);
```

```
  if (adc>512&&adc1>512)
```

```
  { servoA(0);
```

```
    servoB(3); }
```

```
  if (adc<512&&adc1<512)
```

```
  { servoA(0);
```

```
    servoB(-40); }
```

```
  if (adc<512&&adc1>512)
```

```
  {
```

```
    { for (i=0;i<=90;i++)
```

```
      { servoA(i);
```

```
        delay_ms(30); }
```

```
    delay_ms(500);
```

```
  }
```

```
  for (h=90;h>=0;h--)
```

```
  { servoA(h);
```

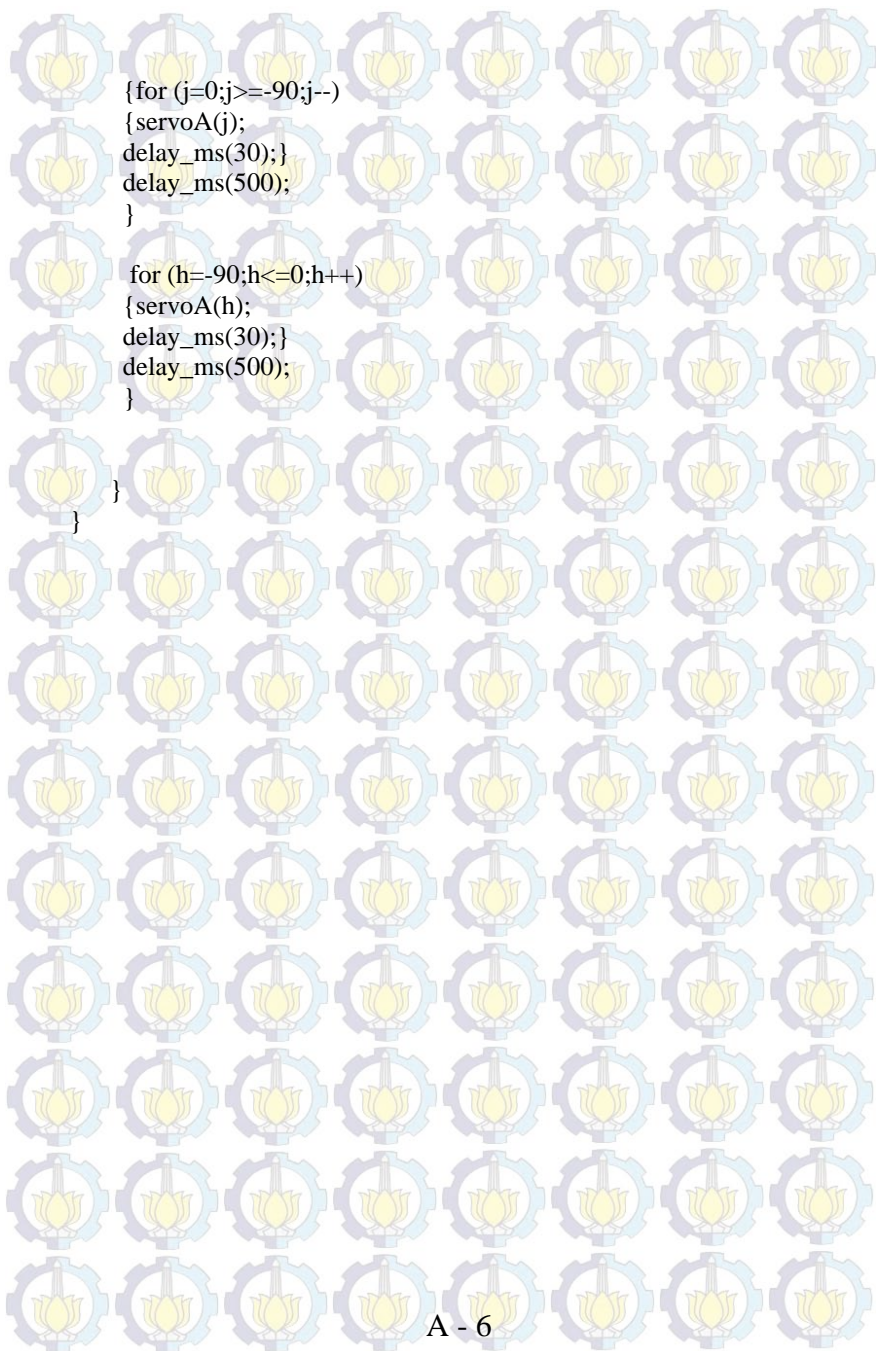
```
    delay_ms(30); }
```

```
  delay_ms(500);
```

```
  }
```

```
  if (adc>512&&adc1<512)
```

```
  {
```



```
{ for (j=0;j<=90;j++)
```

```
{ servoA(j);
```

```
delay_ms(30);}
```

```
delay_ms(500);
```

```
}
```

```
for (h=90;h<=0;h++)
```

```
{ servoA(h);
```

```
delay_ms(30);}
```

```
delay_ms(500);
```

```
}
```

```
}
```

```
}
```



### ***Listing Program CodeWizardAVR Indikator Sinyal***

/\*\*\*\*\*

This program was produced by the  
CodeWizardAVR V2.05.3 Standard  
Automatic Program Generator

© Copyright 1998-2011 Pavel Haiduc, HP InfoTech s.r.l.  
<http://www.hpinfotech.com>

Project :  
Version :  
Date : 21/06/2014  
Author : tyery08  
Company : embeeminded.blogspot.com  
Comments:

Chip type : ATmega16  
Program type : Application  
AVR Core Clock frequency: 12,000000 MHz  
Memory model : Small  
External RAM size : 0  
Data Stack size : 256

\*\*\*\*\*/

```
#include <mega16.h>
#include <stdio.h>
#include <stdlib.h>
#include <delay.h>
```

```
// Alphanumeric LCD functions
#include <alcd.h>
#asm
.equ __lcd_port=0x15 ;PORTC
#endasm
```

```
#define ADC_VREF_TYPE 0x00
char x[16];

int data,data1, y;
```

```

// Read the AD conversion result
unsigned int read_adc(unsigned char adc_input)
{
    ADMUX=adc_input | (ADC_VREF_TYPE & 0xff);
    // Delay needed for the stabilization of the ADC input voltage
    delay_us(10);
    // Start the AD conversion
    ADCSRA|=0x40;
    // Wait for the AD conversion to complete
    while ((ADCSRA & 0x10)==0);
    ADCSRA|=0x10;
    return ADCW;
}

// Declare your global variables here

void main(void)
{
    // Declare your local variables here

    // Input/Output Ports initialization
    // Port A initialization
    // Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In
    Func1=In Func0=In
    // State7=T State6=T State5=T State4=T State3=T State2=T State1=T
    State0=T
    PORTA=0x00;
    DDRA=0x00;

    // Port B initialization
    // Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In
    Func1=In Func0=In
    // State7=T State6=T State5=T State4=T State3=T State2=T State1=T
    State0=T
    PORTB=0x00;
    DDRB=0x00;

    // Port C initialization

```



```

// Func7=Out Func6=Out Func5=Out Func4=Out Func3=Out
Func2=Out Func1=Out Func0=Out
// State7=0 State6=0 State5=0 State4=0 State3=0 State2=0 State1=0
State0=0
PORTC=0x00;
DDRC=0xFF;

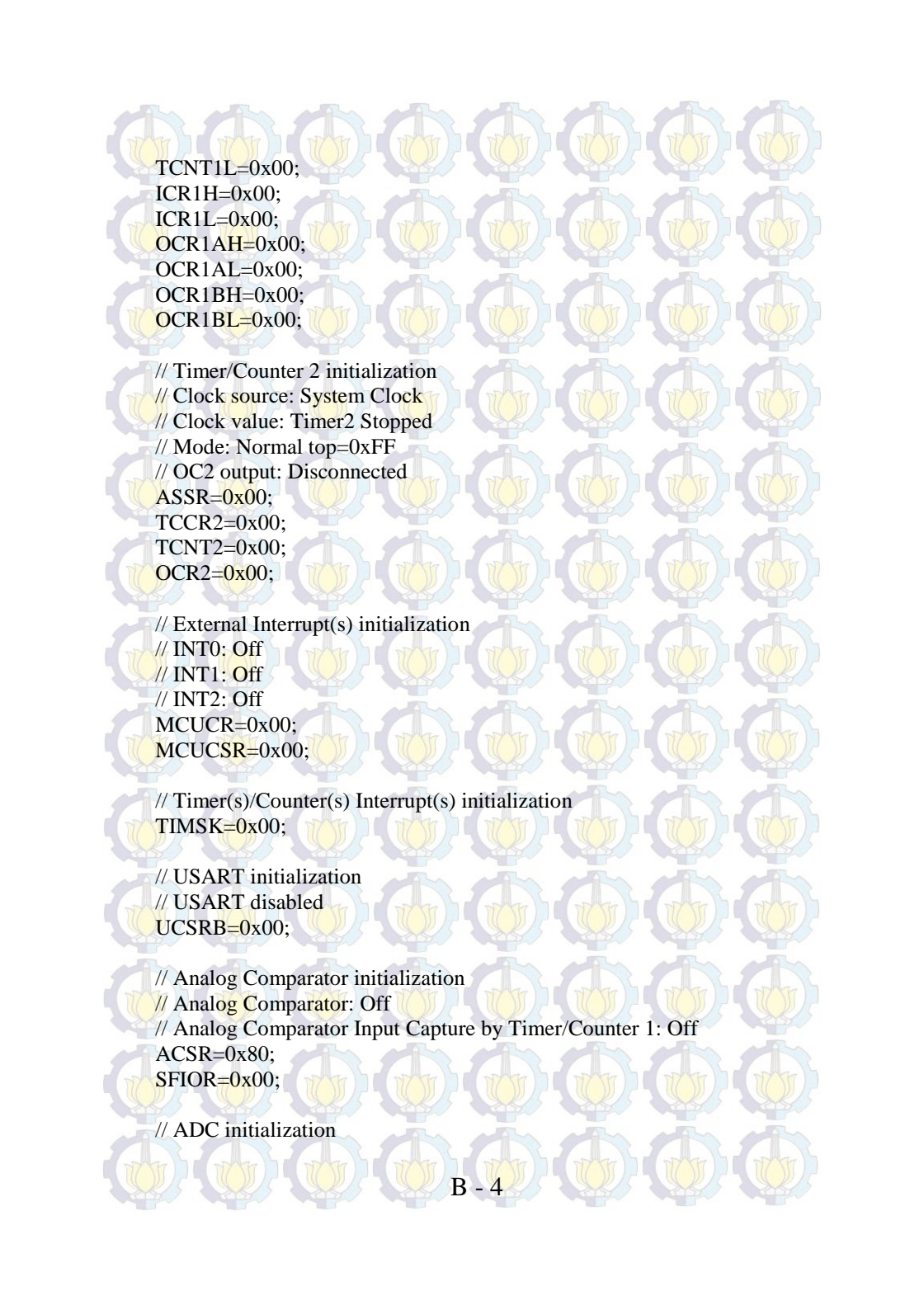
// Port D initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In
Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T
State0=T
PORTD=0x00;
DDRD=0x00;

// Timer/Counter 0 initialization
// Clock source: System Clock
// Clock value: Timer 0 Stopped
// Mode: Normal top=0xFF
// OC0 output: Disconnected
TCCR0=0x00;
TCNT0=0x00;
OCR0=0x00;

// Timer/Counter 1 initialization
// Clock source: System Clock
// Clock value: Timer1 Stopped
// Mode: Normal top=0xFFFF
// OC1A output: Discon.
// OC1B output: Discon.
// Noise Canceler: Off
// Input Capture on Falling Edge
// Timer1 Overflow Interrupt: Off
// Input Capture Interrupt: Off
// Compare A Match Interrupt: Off
// Compare B Match Interrupt: Off
TCCR1A=0x00;
TCCR1B=0x00;
TCNT1H=0x00;

```





```
TCNT1L=0x00;  
ICR1H=0x00;  
ICR1L=0x00;  
OCR1AH=0x00;  
OCR1AL=0x00;  
OCR1BH=0x00;  
OCR1BL=0x00;
```

```
// Timer/Counter 2 initialization  
// Clock source: System Clock  
// Clock value: Timer2 Stopped  
// Mode: Normal top=0xFF  
// OC2 output: Disconnected  
ASSR=0x00;  
TCCR2=0x00;  
TCNT2=0x00;  
OCR2=0x00;
```

```
// External Interrupt(s) initialization  
// INT0: Off  
// INT1: Off  
// INT2: Off  
MCUCR=0x00;  
MCUCSR=0x00;
```

```
// Timer(s)/Counter(s) Interrupt(s) initialization  
TIMSK=0x00;
```

```
// USART initialization  
// USART disabled  
UCSRB=0x00;
```

```
// Analog Comparator initialization  
// Analog Comparator: Off  
// Analog Comparator Input Capture by Timer/Counter 1: Off  
ACSR=0x80;  
SFIOR=0x00;
```

```
// ADC initialization
```

```

// ADC Clock frequency: 375,000 kHz
// ADC Voltage Reference: AREF pin
// ADC Auto Trigger Source: ADC Stopped
ADMUX=ADC_VREF_TYPE & 0xff;
ADCSRA=0x85;

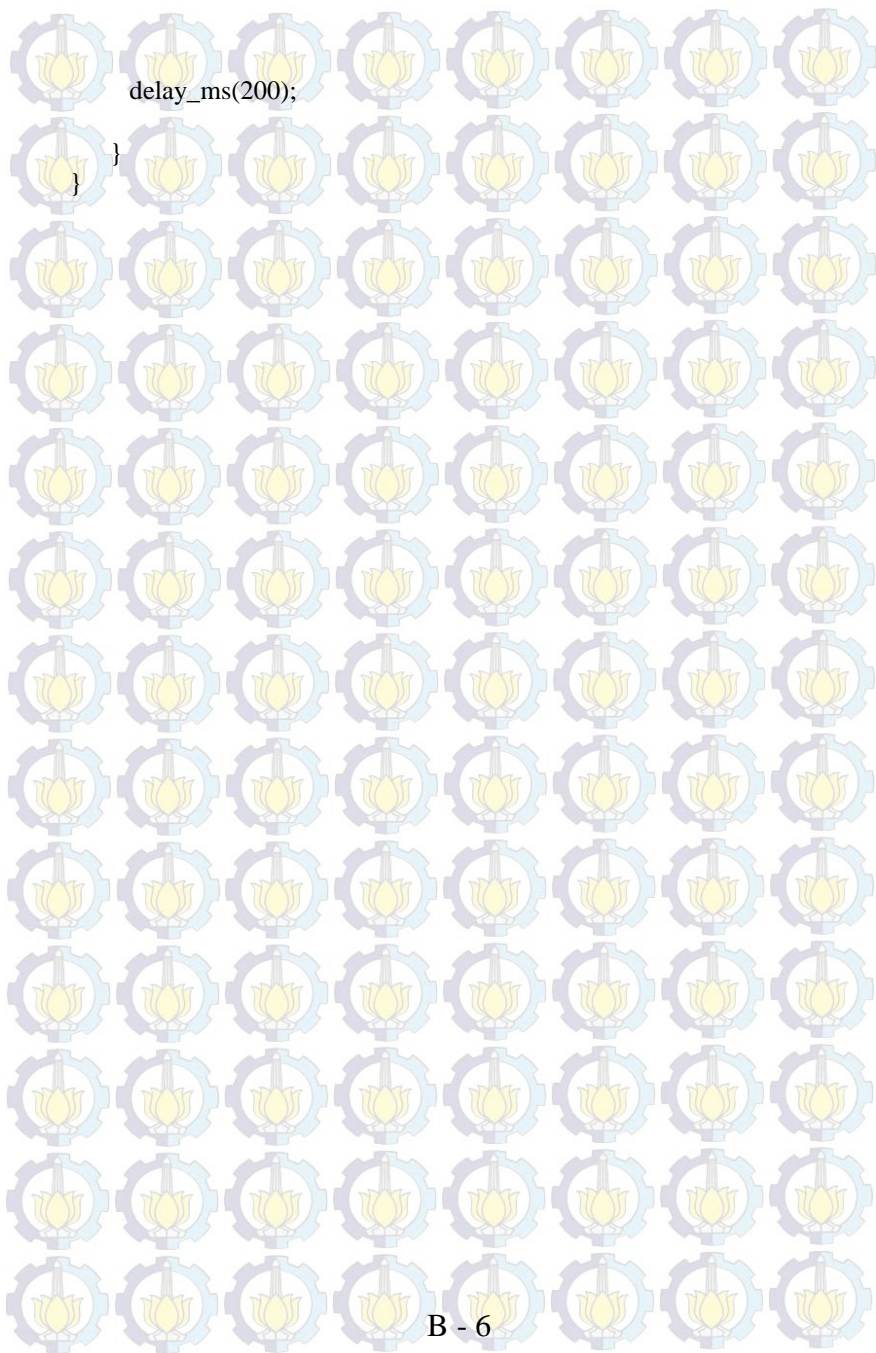
// SPI initialization
// SPI disabled
SPCR=0x00;

// TWI initialization
// TWI disabled
TWCR=0x00;

// Alphanumeric LCD initialization
// Connections are specified in the
// Project|Configure|C Compiler|Libraries|Alphanumeric LCD menu:
// RS - PORTC Bit 0
// RD - PORTC Bit 1
// EN - PORTC Bit 2
// D4 - PORTC Bit 4
// D5 - PORTC Bit 5
// D6 - PORTC Bit 6
// D7 - PORTC Bit 7
// Characters/line: 16
lcd_init(16);

while (1)
{
    data=read_adc(0);
    data1=read_adc(1);
    y=((float)(data-311)*100/450);
    lcd_gotoxy(0,0);
    lcd_putsf("SIGNAL:");
    lcd_gotoxy(11,0);
    lcd_putsf("%");
    sprintf(x,"%i ",y);
    lcd_gotoxy(7,0);
    lcd_puts(x);
}

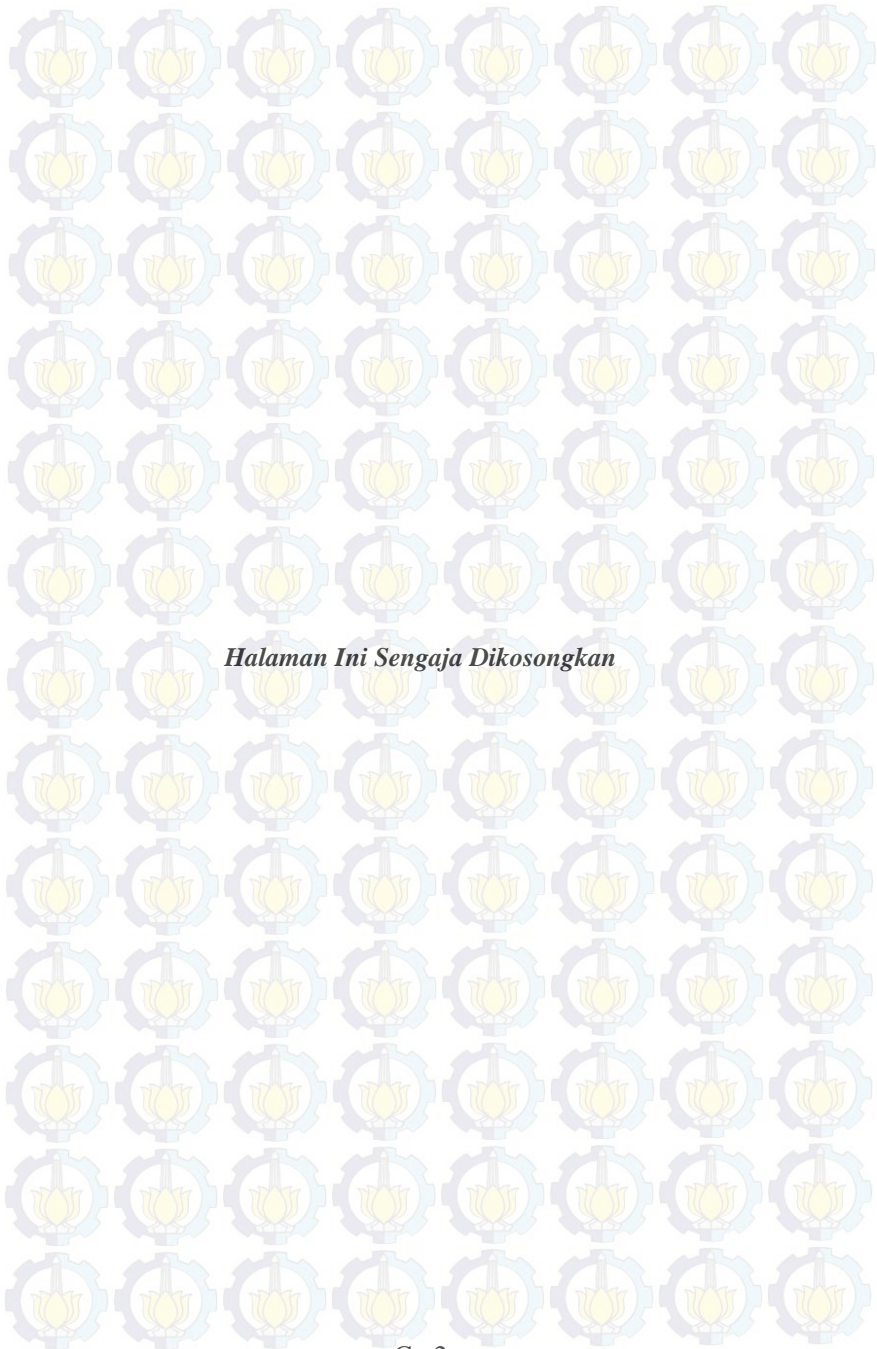
```





## Bentuk Fisik Keseluruhan Sistem





## Bentuk Fisik *Ground Station*

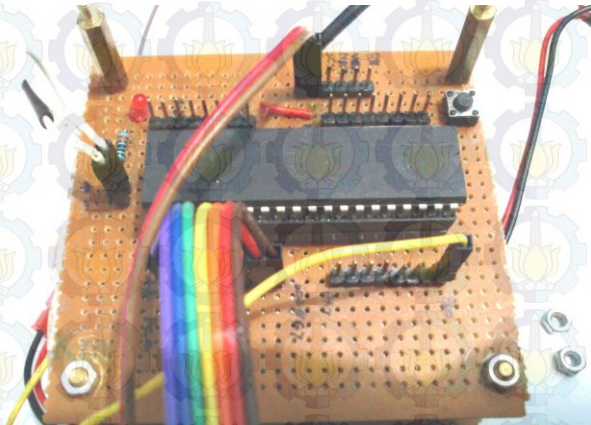
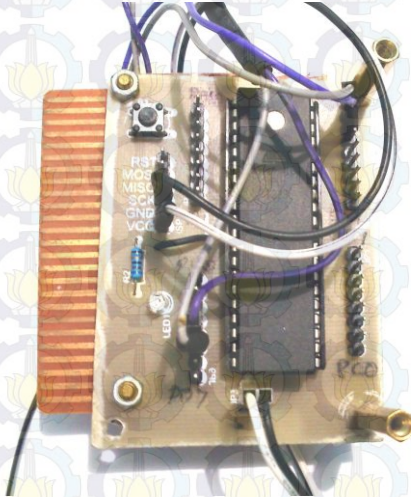


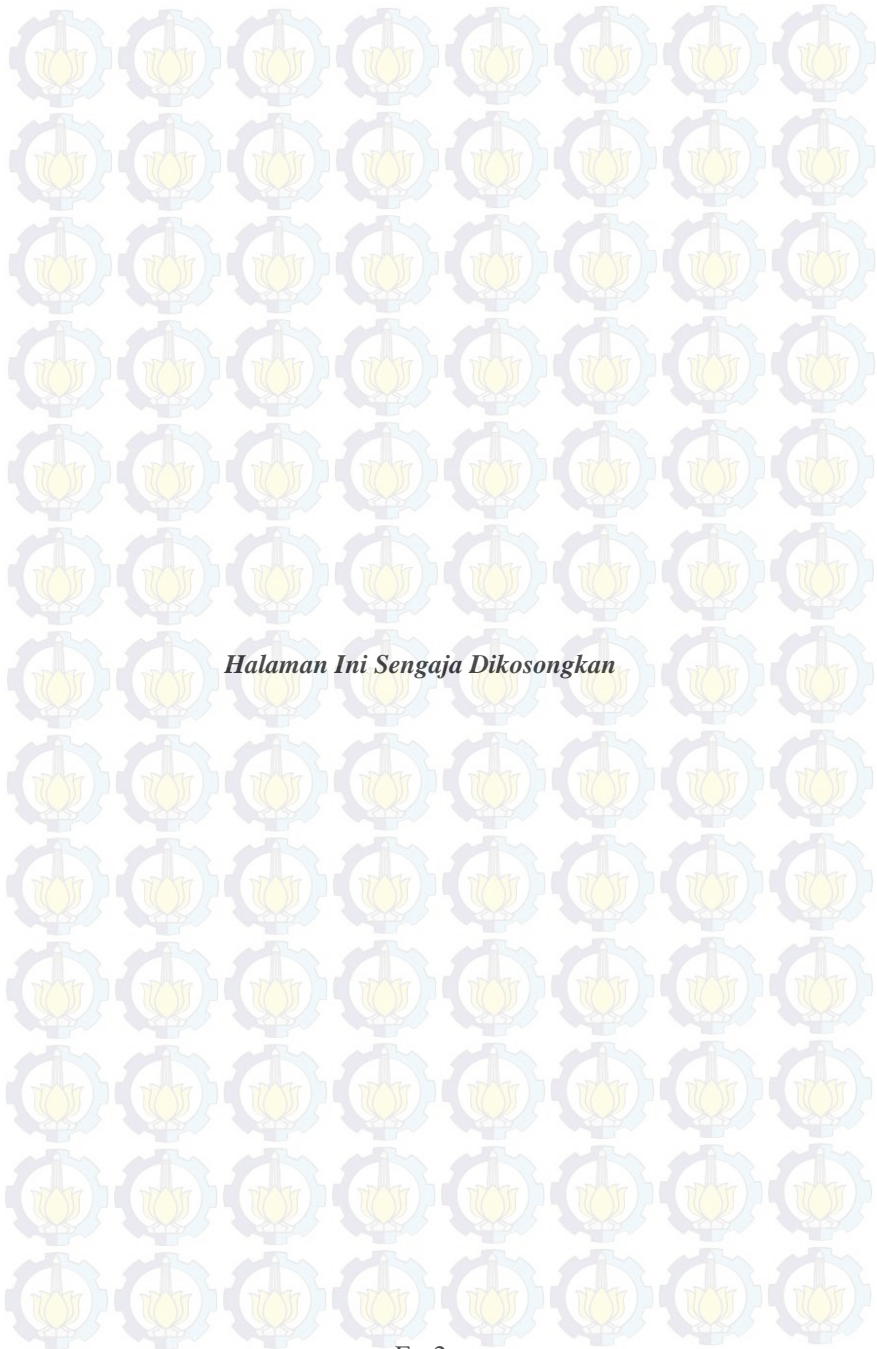




*Halaman Ini Sengaja Dikosongkan*

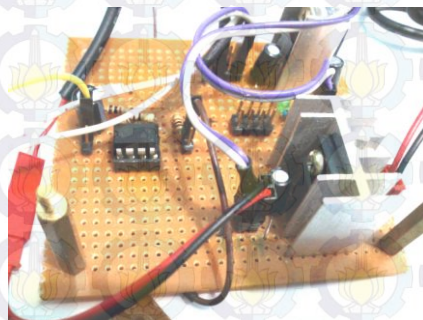
## Bentuk Fisik Mikrokontroler Untuk Pergerakan Kamera dan Indikator Sinyal

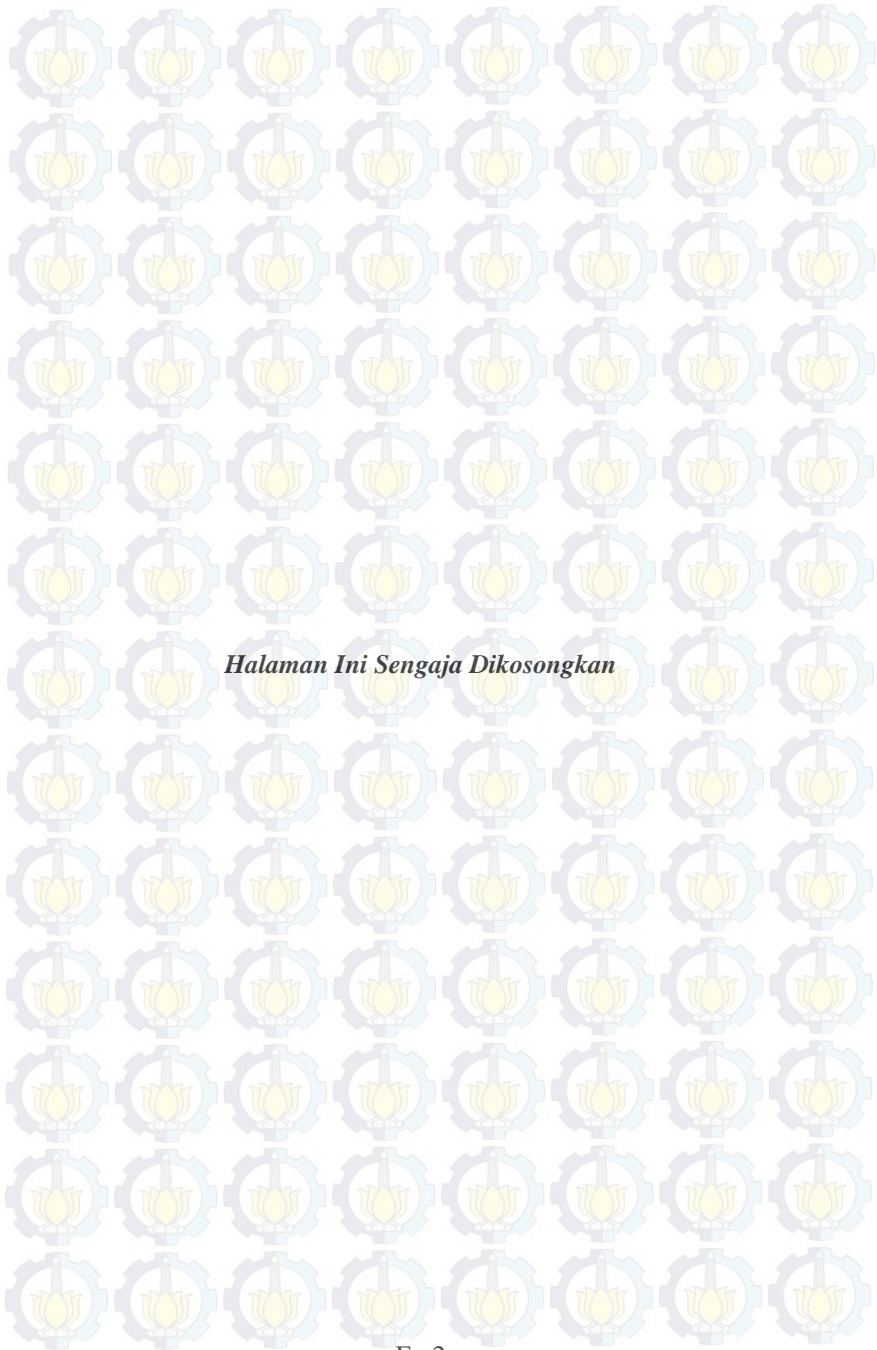






**Bentuk Fisik *OpAmp* Untuk Pergerakan Kamera dan Indikator Sinyal**





## Hasil Gambar Pemantauan







*Halaman Ini Sengaja Dikosongkan*

## Datasheet ATmega16

### Features

- High-performance, Low-power Atmel® AVR® 8-bit Microcontroller
- Advanced RISC Architecture
  - 131 Powerful Instructions – Most Single-clock Cycle Execution
  - 32 x 8 General Purpose Working Registers
  - Fully Static Operation
  - Up to 16 MIPS Throughput at 16 MHz
  - On-chip 2-cycle Multiplier
- High Endurance Non-volatile Memory segments
  - 16 Kbytes of In-System Self-programmable Flash program memory
  - 512 Bytes EEPROM
  - 1 Kbyte Internal SRAM
  - Write/Erase Cycles: 10,000 Flash/100,000 EEPROM
  - Data retention: 20 years at 85°C/100 years at 25°C<sup>(1)</sup>
  - Optional Boot Code Section with Independent Lock Bits
  - In-System Programming by On-chip Boot Program
  - True Read-While-Write Operation
  - Programming Lock for Software Security
- JTAG (IEEE std. 1149.1 Compliant) Interface
  - Boundary-scan Capabilities According to the JTAG Standard
  - Extensive On-chip Debug Support
  - Programming of Flash, EEPROM, Fuses, and Lock Bits through the JTAG Interface
- Peripheral Features
  - Two 8-bit Timer/Counters with Separate Prescalers and Compare Modes
  - One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
  - Real Time Counter with Separate Oscillator
  - Four PWM Channels
  - 8-channel, 10-bit ADC
    - 8 Single-ended Channels
    - 7 Differential Channels in TQFP Package Only
    - 2 Differential Channels with Programmable Gain at 1x, 10x, or 200x
  - Byte-oriented Two-wire Serial Interface
  - Programmable Serial USART
  - Master/Slave SPI Serial Interface
  - Programmable Watchdog Timer with Separate On-chip Oscillator
  - On-chip Analog Comparator
- Special Microcontroller Features
  - Power-on Reset and Programmable Brown-out Detection
  - Internal Calibrated RC Oscillator
  - External and Internal Interrupt Sources
  - Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby and Extended Standby
- I/O and Packages
  - 32 Programmable I/O Lines
  - 40-pin PDIP, 44-lead TQFP, and 44-pad QFN/MLF
- Operating Voltages
  - 2.7V - 5.5V for ATmega16L
  - 4.5V - 5.5V for ATmega16
- Speed Grades
  - 0 - 8 MHz for ATmega16L
  - 0 - 16 MHz for ATmega16
- Power Consumption @ 1 MHz, 3V, and 25°C for ATmega16L
  - Active: 1.1 mA
  - Idle Mode: 0.35 mA
  - Power-down Mode: < 1 µA



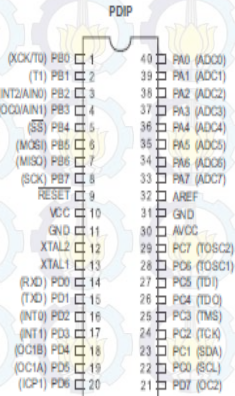
**8-bit AVR®**  
**Microcontroller**  
**with 16K Bytes**  
**In-System**  
**Programmable**  
**Flash**

**ATmega16**  
**ATmega16L**

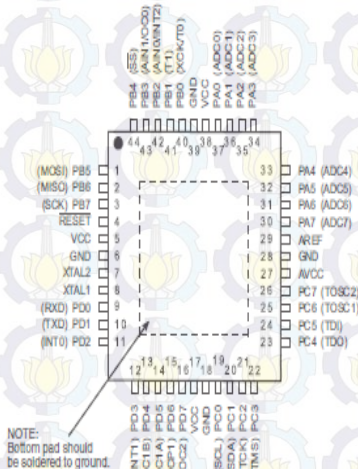
Rev. 2466T-AVR-07/10

## Pin Configurations

Figure 1. Pinout ATmega16



TQFP/QFN/MLF



## Disclaimer

Typical values contained in this datasheet are based on simulations and characterization of other AVR microcontrollers manufactured on the same process technology. Min and Max values will be available after the device is characterized.



## Overview

The ATmega16 is a low-power CMOS 8-bit microcontroller based on the AVR enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega16 achieves throughputs approaching 1 MIPS per MHz allowing the system designer to optimize power consumption versus processing speed.

## Block Diagram

Figure 2. Block Diagram

The block diagram illustrates the internal architecture of the ATmega16 microcontroller. It is divided into two main sections: the AVR CPU and the peripheral components. The AVR CPU section includes the Program Counter, Program Flash, Instruction Register, Instruction Decoder, Control Lines, Status Register, ALU, and General Purpose Registers (X, Y, Z). The peripheral components section includes the Oscillator, Timers/Counters, TWI, SPI, USART, and various interfaces (PORTA, PORTB, PORTC, PORTD). The diagram shows the connections between these components, including the VCC and GND power supply, the AVCC and AREF analog supply, and the RESET pin. The diagram also shows the connection to external components like the Oscillator, Timers/Counters, TWI, SPI, USART, and various interfaces (PORTA, PORTB, PORTC, PORTD).

## Overview

The ATmega16 is a low-power CMOS 8-bit microcontroller based on the AVR enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega16 achieves throughputs approaching 1 MIPS per MHz allowing the system designer to optimize power consumption versus processing speed.

## Block Diagram

Figure 2. Block Diagram

The block diagram illustrates the internal architecture of the ATmega16 microcontroller. It is organized into four main quadrants around a central vertical bus. The top-left quadrant contains the PORTA DRIVERS/BUFFERS, PORTA DIGITAL INTERFACE, and PORTA DIGITAL INTERFACE. The top-right quadrant contains the PORTC DRIVERS/BUFFERS, PORTC DIGITAL INTERFACE, and PORTC DIGITAL INTERFACE. The bottom-left quadrant contains the PORTD DRIVERS/BUFFERS, PORTD DIGITAL INTERFACE, and PORTD DIGITAL INTERFACE. The bottom-right quadrant contains the PORTE DRIVERS/BUFFERS, PORTE DIGITAL INTERFACE, and PORTE DIGITAL INTERFACE. The central vertical bus connects these peripheral blocks to the core AVR CPU and other internal components. The AVR CPU core includes the PROGRAM COUNTER, PROGRAM FLASH, INSTRUCTION REGISTER, INSTRUCTION DECODER, CONTROL LINES, STATUS REGISTER, STACK POINTER, SRAM, GENERAL PURPOSE REGISTERS (X, Y, Z), ALU, and AVR CPU. Other internal components include the MUX & ADC, ADC INTERFACE, TWI, TIMERS/COUNTERS, OSCILLATOR, INTERNAL OSCILLATOR, WATCHDOG TIMER, MCU CTRL. & TIMING, INTERRUPT UNIT, INTERNAL CALIBRATED OSCILLATOR, EEPROM, USART, SPI, and COMP INTERFACE. The diagram also shows external connections for VCC, GND, AVCC, AREF, XTAL1, XTAL2, and RESET.

## Overview

The ATmega16 is a low-power CMOS 8-bit microcontroller based on the AVR enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega16 achieves throughputs approaching 1 MIPS per MHz allowing the system designer to optimize power consumption versus processing speed.

## Block Diagram

Figure 2. Block Diagram

The block diagram illustrates the internal architecture of the ATmega16 microcontroller. It is organized into several main functional blocks:

- AVR CPU Core:** The central processing unit, containing the Program Counter, Program Flash, Instruction Register, Instruction Decoder, Control Lines, Status Register, Stack Pointer, SRAM, General Purpose Registers (X, Y, Z), and the ALU.
- Peripheral Modules:** Various modules connected to the CPU core, including Timers/Counters, Oscillator, Watchdog Timer, MCU CTRL & TIMING, Interrupt Unit, Internal Calibrated Oscillator, EEPROM, USART, SPI, and COMP INTERFACE.
- Port Modules:** PORTA and PORTC, each consisting of PORTA/PORTC Drivers/Buffers, PORTA/PORTC Digital Interfaces, and PORTA/PORTC Registers.
- Power and Reset:** Power pins VCC, GND, AVCC, and AREF. External pins include XTAL1, XTAL2, and RESET.

The diagram shows the interconnections between these blocks, highlighting the flow of data and control signals throughout the microcontroller.

## Overview

The ATmega16 is a low-power CMOS 8-bit microcontroller based on the AVR enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega16 achieves throughputs approaching 1 MIPS per MHz allowing the system designer to optimize power consumption versus processing speed.

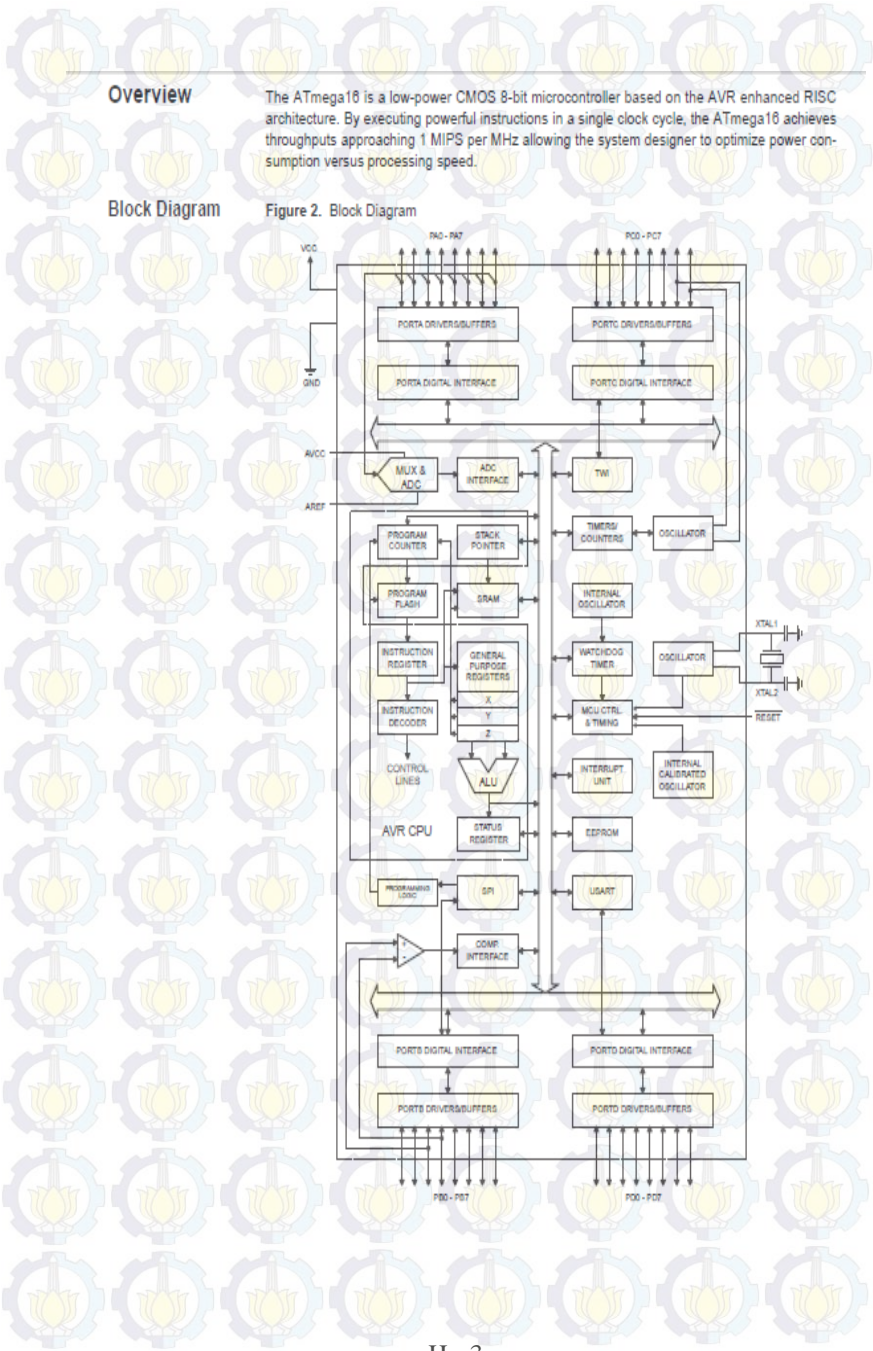
## Block Diagram

Figure 2. Block Diagram

The block diagram illustrates the internal architecture of the ATmega16 microcontroller. It is organized into several main functional blocks:

- AVR CPU Core:** The central processing unit, containing the Program Counter, Program Flash, Instruction Register, Instruction Decoder, Control Lines, Status Register, Stack Pointer, SRAM, General Purpose Registers (X, Y, Z), and the ALU.
- Peripheral Modules:** Various modules connected to the CPU core, including Timers/Counters, Oscillator, Watchdog Timer, MCU CTRL & TIMING, Interrupt Unit, Internal Calibrated Oscillator, EEPROM, USART, SPI, and COMP INTERFACE.
- Port Modules:** PORTA and PORTC, each consisting of PORTA/PORTC Drivers/Buffers, PORTA/PORTC Digital Interfaces, and PORTA/PORTC Registers.
- Power and Reset:** Power pins VCC, GND, AVCC, and AREF. External pins include XTAL1, XTAL2, and RESET.

The diagram shows the interconnections between these blocks, highlighting the flow of data and control signals throughout the microcontroller.



The AVR core combines a rich instruction set with 32 general purpose working registers. All the 32 registers are directly connected to the Arithmetic Logic Unit (ALU), allowing two independent registers to be accessed in one single instruction executed in one clock cycle. The resulting architecture is more code efficient while achieving throughputs up to ten times faster than conventional CISC microcontrollers.

The ATmega16 provides the following features: 16 Kbytes of In-System Programmable Flash Program memory with Read-While-Write capabilities, 512 bytes EEPROM, 1 Kbyte SRAM, 32 general purpose I/O lines, 32 general purpose working registers, a JTAG interface for Boundary-scan, On-chip Debugging support and programming, three flexible Timer/Counters with compare modes, Internal and External Interrupts, a serial programmable USART, a byte oriented Two-wire Serial Interface, an 8-channel, 10-bit ADC with optional differential input stage with programmable gain (TQFP package only), a programmable Watchdog Timer with Internal Oscillator, an SPI serial port, and six software selectable power saving modes. The Idle mode stops the CPU while allowing the USART, Two-wire interface, A/D Converter, SRAM, Timer/Counters, SPI port, and interrupt system to continue functioning. The Power-down mode saves the register contents but freezes the Oscillator, disabling all other chip functions until the next External Interrupt or Hardware Reset. In Power-save mode, the Asynchronous Timer continues to run, allowing the user to maintain a timer base while the rest of the device is sleeping. The ADC Noise Reduction mode stops the CPU and all I/O modules except Asynchronous Timer and ADC, to minimize switching noise during ADC conversions. In Standby mode, the crystal/resonator Oscillator is running while the rest of the device is sleeping. This allows very fast start-up combined with low-power consumption. In Extended Standby mode, both the main Oscillator and the Asynchronous Timer continue to run.

The device is manufactured using Atmel's high density nonvolatile memory technology. The On-chip ISP Flash allows the program memory to be reprogrammed in-system through an SPI serial interface, by a conventional nonvolatile memory programmer, or by an On-chip Boot program running on the AVR core. The boot program can use any interface to download the application program in the Application Flash memory. Software in the Boot Flash section will continue to run while the Application Flash section is updated, providing true Read-While-Write operation. By combining an 8-bit RISC CPU with In-System Self-Programmable Flash on a monolithic chip, the Atmel ATmega16 is a powerful microcontroller that provides a highly-flexible and cost-effective solution to many embedded control applications.

The ATmega16 AVR is supported with a full suite of program and system development tools including: C compilers, macro assemblers, program debugger/simulators, in-circuit emulators, and evaluation kits.

## Pin Descriptions

VCC

Digital supply voltage.

GND

Ground.

Port A (PA7..PA0)

Port A serves as the analog inputs to the A/D Converter.

Port A also serves as an 8-bit bi-directional I/O port, if the A/D Converter is not used. Port pins can provide internal pull-up resistors (selected for each bit). The Port A output buffers have symmetrical drive characteristics with both high sink and source capability. When pins PA0 to PA7 are used as inputs and are externally pulled low, they will source current if the internal pull-up resistors are activated. The Port A pins are tri-stated when a reset condition becomes active, even if the clock is not running.

#### Port B (PB7..PB0)

Port B is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port B output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port B pins that are externally pulled low will source current if the pull-up resistors are activated. The Port B pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Port B also serves the functions of various special features of the ATmega16 as listed on [page 58](#).

#### Port C (PC7..PC0)

Port C is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port C output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port C pins that are externally pulled low will source current if the pull-up resistors are activated. The Port C pins are tri-stated when a reset condition becomes active, even if the clock is not running. If the JTAG interface is enabled, the pull-up resistors on pins PC5(TDI), PC3(TMS) and PC2(TCK) will be activated even if a reset occurs.

Port C also serves the functions of the JTAG interface and other special features of the ATmega16 as listed on [page 61](#).

#### Port D (PD7..PD0)

Port D is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port D output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port D pins that are externally pulled low will source current if the pull-up resistors are activated. The Port D pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Port D also serves the functions of various special features of the ATmega16 as listed on [page 63](#).

#### RESET

Reset Input. A low level on this pin for longer than the minimum pulse length will generate a reset, even if the clock is not running. The minimum pulse length is given in [Table 15 on page 38](#). Shorter pulses are not guaranteed to generate a reset.

#### XTAL1

Input to the inverting Oscillator amplifier and input to the internal clock operating circuit.

#### XTAL2

Output from the inverting Oscillator amplifier.

#### AVCC

AVCC is the supply voltage pin for Port A and the A/D Converter. It should be externally connected to  $V_{CC}$ , even if the ADC is not used. If the ADC is used, it should be connected to  $V_{CC}$  through a low-pass filter.

#### AREF

AREF is the analog reference pin for the A/D Converter.



## External Interrupts

The External Interrupts are triggered by the INT0, INT1, and INT2 pins. Observe that, if enabled, the interrupts will trigger even if the INT0.2 pins are configured as outputs. This feature provides a way of generating a software interrupt. The external interrupts can be triggered by a falling or rising edge or a low level (INT2 is only an edge triggered interrupt). This is set up as indicated in the specification for the MCU Control Register – MCUCR – and MCU Control and Status Register – MCUCSR. When the external interrupt is enabled and is configured as level triggered (only INT0/INT1), the interrupt will trigger as long as the pin is held low. Note that recognition of falling or rising edge interrupts on INT0 and INT1 requires the presence of an I/O clock, described in ["Clock Systems and their Distribution" on page 24](#). Low level interrupts on INT0/INT1 and the edge interrupt on INT2 are detected asynchronously. This implies that these interrupts can be used for waking the part also from sleep modes other than Idle mode. The I/O clock is halted in all sleep modes except Idle mode.

Note that if a level triggered interrupt is used for wake-up from Power-down mode, the changed level must be held for some time to wake up the MCU. This makes the MCU less sensitive to noise. The changed level is sampled twice by the Watchdog Oscillator clock. The period of the Watchdog Oscillator is 1  $\mu$ s (nominal) at 5.0V and 25°C. The frequency of the Watchdog Oscillator is voltage dependent as shown in ["Electrical Characteristics" on page 291](#). The MCU will wake up if the input has the required level during this sampling or if it is held until the end of the start-up time. The start-up time is defined by the SUT Fuses as described in ["System Clock and Clock Options" on page 24](#). If the level is sampled twice by the Watchdog Oscillator clock but disappears before the end of the start-up time, the MCU will still wake up, but no interrupt will be generated. The required level must be held long enough for the MCU to complete the wake up to trigger the level interrupt.

## MCU Control Register – MCUCR

The MCU Control Register contains control bits for interrupt sense control and general MCU functions.

Bit	7	6	5	4	3	2	1	0	
	SM2	SE	SM1	SM0	ISC11	ISC10	ISC01	ISC00	MCUCR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- Bit 3 – ISC11, ISC10: Interrupt Sense Control 1 Bit 1 and Bit 0

The External Interrupt 1 is activated by the external pin INT1 if the SREG I-bit and the corresponding interrupt mask in the GICR are set. The level and edges on the external INT1 pin that activate the interrupt are defined in [Table 34](#). The value on the INT1 pin is sampled before detecting edges. If edge or toggle interrupt is selected, pulses that last longer than one clock period will generate an interrupt. Shorter pulses are not guaranteed to generate an interrupt. If low level interrupt is selected, the low level must be held until the completion of the currently executing instruction to generate an interrupt.

Table 34. Interrupt 1 Sense Control

ISC11	ISC10	Description
0	0	The low level of INT1 generates an interrupt request.
0	1	Any logical change on INT1 generates an interrupt request.
1	0	The falling edge of INT1 generates an interrupt request.
1	1	The rising edge of INT1 generates an interrupt request.

## External Interrupts

The External Interrupts are triggered by the INT0, INT1, and INT2 pins. Observe that, if enabled, the interrupts will trigger even if the INT0..2 pins are configured as outputs. This feature provides a way of generating a software interrupt. The external interrupts can be triggered by a falling or rising edge or a low level (INT2 is only an edge triggered interrupt). This is set up as indicated in the specification for the MCU Control Register – MCUCR – and MCU Control and Status Register – MCUCSR. When the external interrupt is enabled and is configured as level triggered (only INT0/INT1), the interrupt will trigger as long as the pin is held low. Note that recognition of falling or rising edge interrupts on INT0 and INT1 requires the presence of an I/O clock, described in “Clock Systems and their Distribution” on page 24. Low level interrupts on INT0/INT1 and the edge interrupt on INT2 are detected asynchronously. This implies that these interrupts can be used for waking the part also from sleep modes other than Idle mode. The I/O clock is halted in all sleep modes except Idle mode.

Note that if a level triggered interrupt is used for wake-up from Power-down mode, the changed level must be held for some time to wake up the MCU. This makes the MCU less sensitive to noise. The changed level is sampled twice by the Watchdog Oscillator clock. The period of the Watchdog Oscillator is 1  $\mu$ s (nominal) at 5.0V and 25°C. The frequency of the Watchdog Oscillator is voltage dependent as shown in “Electrical Characteristics” on page 291. The MCU will wake up if the input has the required level during this sampling or if it is held until the end of the start-up time. The start-up time is defined by the SUT Fuses as described in “System Clock and Clock Options” on page 24. If the level is sampled twice by the Watchdog Oscillator clock but disappears before the end of the start-up time, the MCU will still wake up, but no interrupt will be generated. The required level must be held long enough for the MCU to complete the wake up to trigger the level interrupt.

### MCU Control Register – MCUCR

The MCU Control Register contains control bits for interrupt sense control and general MCU functions.

Bit	7	6	5	4	3	2	1	0	
	SM2	SE	SM1	SM0	ISC11	ISC10	ISC01	ISC00	MCUCR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

#### • Bit 3, 2 – ISC11, ISC10: Interrupt Sense Control 1 Bit 1 and Bit 0

The External Interrupt 1 is activated by the external pin INT1 if the SREG I-bit and the corresponding interrupt mask in the GICR are set. The level and edges on the external INT1 pin that activate the interrupt are defined in Table 34. The value on the INT1 pin is sampled before detecting edges. If edge or toggle interrupt is selected, pulses that last longer than one clock period will generate an interrupt. Shorter pulses are not guaranteed to generate an interrupt. If low level interrupt is selected, the low level must be held until the completion of the currently executing instruction to generate an interrupt.

Table 34. Interrupt 1 Sense Control

ISC11	ISC10	Description
0	0	The low level of INT1 generates an interrupt request.
0	1	Any logical change on INT1 generates an interrupt request.
1	0	The falling edge of INT1 generates an interrupt request.
1	1	The rising edge of INT1 generates an interrupt request.



# Timer/Counter1 Control Register B – TCCR1B

Bit	7	6	5	4	3	2	1	0	TCCR1B
	ICNC1	ICES1	–	WGM13	WGM12	CS12	CS11	CS10	
Read/Write	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

## • Bit 7 – ICNC1: Input Capture Noise Canceler

Setting this bit (to one) activates the Input Capture Noise Canceler. When the Noise Canceler is activated, the input from the Input Capture Pin (ICP1) is filtered. The filter function requires four successive equal valued samples of the ICP1 pin for changing its output. The Input Capture is therefore delayed by four Oscillator cycles when the Noise Canceler is enabled.

## • Bit 6 – ICES1: Input Capture Edge Select

This bit selects which edge on the Input Capture Pin (ICP1) that is used to trigger a capture event. When the ICES1 bit is written to zero, a falling (negative) edge is used as trigger, and when the ICES1 bit is written to one, a rising (positive) edge will trigger the capture.

When a capture is triggered according to the ICES1 setting, the counter value is copied into the Input Capture Register (ICR1). The event will also set the Input Capture Flag (ICF1), and this can be used to cause an Input Capture Interrupt, if this interrupt is enabled.

When the ICR1 is used as TOP value (see description of the WGM13:0 bits located in the TCCR1A and the TCCR1B Register), the ICP1 is disconnected and consequently the Input Capture function is disabled.

## • Bit 5 – Reserved Bit

This bit is reserved for future use. For ensuring compatibility with future devices, this bit must be written to zero when TCCR1B is written.

## • Bit 4:3 – WGM13:2: Waveform Generation Mode

See TCCR1A Register description.

## • Bit 2:0 – CS12:0: Clock Select

The three Clock Select bits select the clock source to be used by the Timer/Counter, see [Figure 49](#) and [Figure 50](#).

Table 48. Clock Select Bit Description

CS12	CS11	CS10	Description
0	0	0	No clock source (Timer/Counter stopped).
0	0	1	$clk_{ICP}/1$ (No prescaling)
0	1	0	$clk_{ICP}/8$ (From prescaler)
0	1	1	$clk_{ICP}/64$ (From prescaler)
1	0	0	$clk_{ICP}/256$ (From prescaler)
1	0	1	$clk_{ICP}/1024$ (From prescaler)
1	1	0	External clock source on T1 pin. Clock on falling edge.
1	1	1	External clock source on T1 pin. Clock on rising edge.



### Timer/Counter1 – TCNT1H and TCNT1L

Bit	7	6	5	4	3	2	1	0
	TCNT1[15:8]							
	TCNT1[7:0]							
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value	0	0	0	0	0	0	0	0

The two *Timer/Counter* I/O locations (TCNT1H and TCNT1L, combined TCNT1) give direct access, both for read and for write operations, to the Timer/Counter unit 16-bit counter. To ensure that both the high and Low bytes are read and written simultaneously when the CPU accesses these registers, the access is performed using an 8-bit temporary High Byte Register (TEMP). This temporary register is shared by all the other 16-bit registers. See ["Accessing 16-bit Registers"](#) on page 92.

Modifying the counter (TCNT1) while the counter is running introduces a risk of missing a compare match between TCNT1 and one of the OCR1x Registers.

Writing to the TCNT1 Register blocks (removes) the compare match on the following timer clock for all compare units.

### Output Compare Register 1 A – OCR1AH and OCR1AL

Bit	7	6	5	4	3	2	1	0
	OCR1A[15:8]							
	OCR1A[7:0]							
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value	0	0	0	0	0	0	0	0

### Output Compare Register 1 B – OCR1BH and OCR1BL

Bit	7	6	5	4	3	2	1	0
	OCR1B[15:8]							
	OCR1B[7:0]							
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value	0	0	0	0	0	0	0	0

The Output Compare Registers contain a 16-bit value that is continuously compared with the counter value (TCNT1). A match can be used to generate an output compare interrupt, or to generate a waveform output on the OC1x pin.

The Output Compare Registers are 16-bit in size. To ensure that both the high and Low bytes are written simultaneously when the CPU writes to these registers, the access is performed using an 8-bit temporary High Byte Register (TEMP). This temporary register is shared by all the other 16-bit registers. See ["Accessing 16-bit Registers"](#) on page 92.

### Input Capture Register 1 – ICR1H and ICR1L

Bit	7	6	5	4	3	2	1	0
	ICR1[15:8]							
	ICR1[7:0]							
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value	0	0	0	0	0	0	0	0

## Analog to Digital Converter

### Features

- 10-bit Resolution
- 0.5 LSB Integral Non-linearity
- $\pm 2$  LSB Absolute Accuracy
- 13  $\mu$ s-260  $\mu$ s Conversion Time
- Up to 15 kSPS at Maximum Resolution
- 8 Multiplexed Single Ended Input Channels
- 7 Differential Input Channels
- 2 Differential Input Channels with Optional Gain of 10x and 200x
- Optional Left adjustment for ADC Result Readout
- 0 -  $V_{CC}$  ADC Input Voltage Range
- Selectable 2.56V ADC Reference Voltage
- Free Running or Single Conversion Mode
- ADC Start Conversion by Auto Triggering on Interrupt Sources
- Interrupt on ADC Conversion Complete
- Sleep Mode Noise Canceler

The ATmega16 features a 10-bit successive approximation ADC. The ADC is connected to an 8-channel Analog Multiplexer which allows 8 single-ended voltage inputs constructed from the pins of Port A. The single-ended voltage inputs refer to 0V (GND).

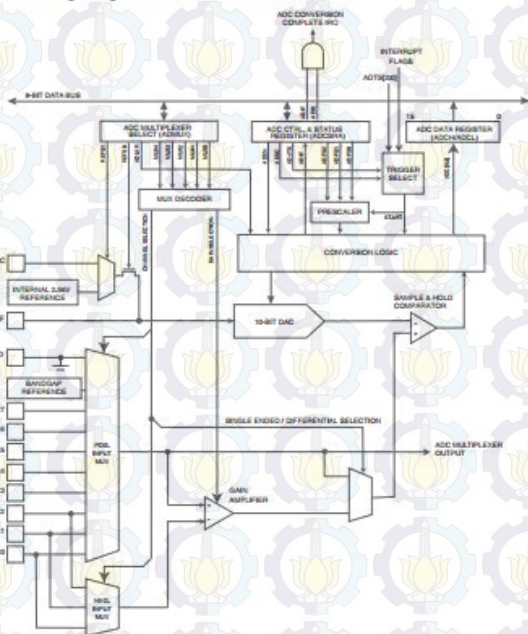
The device also supports 16 differential voltage input combinations. Two of the differential inputs (ADC1, ADC0 and ADC3, ADC2) are equipped with a programmable gain stage, providing amplification steps of 0 dB (1x), 20 dB (10x), or 46 dB (200x) on the differential input voltage before the A/D conversion. Seven differential analog input channels share a common negative terminal (ADC1), while any other ADC input can be selected as the positive input terminal. If 1x or 10x gain is used, 8-bit resolution can be expected. If 200x gain is used, 7-bit resolution can be expected.

The ADC contains a Sample and Hold circuit which ensures that the input voltage to the ADC is held at a constant level during conversion. A block diagram of the ADC is shown in [Figure 98](#).

The ADC has a separate analog supply voltage pin, AVCC. AVCC must not differ more than  $\pm 0.3$ V from  $V_{CC}$ . See the paragraph "ADC Noise Canceler" on page 211 on how to connect this pin.

Internal reference voltages of nominally 2.56V or AVCC are provided On-chip. The voltage reference may be externally decoupled at the AREF pin by a capacitor for better noise performance.

Figure 98. Analog to Digital Converter Block Schematic



## Operation

The ADC converts an analog input voltage to a 10-bit digital value through successive approximation. The minimum value represents GND and the maximum value represents the voltage on the AREF pin minus 1 LSB. Optionally, AVCC or an internal 2.56V reference voltage may be connected to the AREF pin by writing to the REFSn bits in the ADMUX Register. The internal voltage reference may thus be decoupled by an external capacitor at the AREF pin to improve noise immunity.

The analog input channel and differential gain are selected by writing to the MUX bits in ADMUX. Any of the ADC input pins, as well as GND and a fixed bandgap voltage reference, can be selected as single ended inputs to the ADC. A selection of ADC input pins can be selected as positive and negative inputs to the differential gain amplifier.

If differential channels are selected, the differential gain stage amplifies the voltage difference between the selected input channel pair by the selected gain factor. This amplified value then

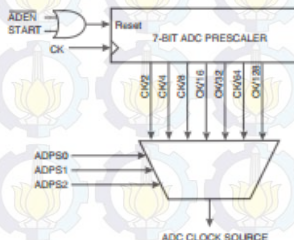


Using the ADC Interrupt Flag as a trigger source makes the ADC start a new conversion as soon as the ongoing conversion has finished. The ADC then operates in Free Running mode, constantly sampling and updating the ADC Data Register. The first conversion must be started by writing a logical one to the ADSC bit in ADCSRA. In this mode the ADC will perform successive conversions independently of whether the ADC Interrupt Flag, ADIF is cleared or not.

If Auto Triggering is enabled, single conversions can be started by writing ADSC in ADCSRA to one. ADSC can also be used to determine if a conversion is in progress. The ADSC bit will be read as one during a conversion, independently of how the conversion was started.

## Prescaling and Conversion Timing

Figure 100. ADC Prescaler



By default, the successive approximation circuitry requires an input clock frequency between 50 kHz and 200 kHz to get maximum resolution. If a lower resolution than 10 bits is needed, the input clock frequency to the ADC can be higher than 200 kHz to get a higher sample rate.

The ADC module contains a prescaler, which generates an acceptable ADC clock frequency from any CPU frequency above 100 kHz. The prescaling is set by the ADPS bits in ADCSRA. The prescaler starts counting from the moment the ADC is switched on by setting the ADEN bit in ADCSRA. The prescaler keeps running for as long as the ADEN bit is set, and is continuously reset when ADEN is low.

When initiating a single ended conversion by setting the ADSC bit in ADCSRA, the conversion starts at the following rising edge of the ADC clock cycle. See ["Differential Gain Channels" on page 209](#) for details on differential conversion timing.

A normal conversion takes 13 ADC clock cycles. The first conversion after the ADC is switched on (ADEN in ADCSRA is set) takes 25 ADC clock cycles in order to initialize the analog circuitry.

The actual sample-and-hold takes place 1.5 ADC clock cycles after the start of a normal conversion and 13.5 ADC clock cycles after the start of a first conversion. When a conversion is complete, the result is written to the ADC Data Registers, and ADIF is set. In single conversion mode, ADSC is cleared simultaneously. The software may then set ADSC again, and a new conversion will be initiated on the first rising ADC clock edge.

When Auto Triggering is used, the prescaler is reset when the trigger event occurs. This assures a fixed delay from the trigger event to the start of conversion. In this mode, the sample-and-hold takes place 2 ADC clock cycles after the rising edge on the trigger source signal. Three additional CPU clock cycles are used for synchronization logic. When using Differential mode, along with Auto triggering from a source other than the ADC Conversion Complete, each conversion

## -Datasheet OpAmp 358

# LM258, LM358, LM358A, LM2904, LM2904A, LM2904V, NCV2904, NCV2904V

## Single Supply Dual Operational Amplifiers

Utilizing the circuit designs perfected for Quad Operational Amplifiers, these dual operational amplifiers feature low power drain, a common mode input voltage range extending to ground/ $V_{EE}$ , and single supply or split supply operation. The LM358 series is equivalent to one-half of an LM324.

These amplifiers have several distinct advantages over standard operational amplifier types in single supply applications. They can operate at supply voltages as low as 3.0 V or as high as 32 V, with quiescent currents about one-fifth of those associated with the MC1741 (on a per amplifier basis). The common mode input range includes the negative supply, thereby eliminating the necessity for external biasing components in many applications. The output voltage range also includes the negative power supply voltage.

### Features

- Short Circuit Protected Outputs
- True Differential Input Stage
- Single Supply Operation: 3.0 V to 32 V
- Low Input Bias Currents
- Internally Compensated
- Common Mode Range Extends to Negative Supply
- Single and Split Supply Operation
- ESD Clamps on the Inputs Increase Ruggedness of the Device without Affecting Operation
- NCV Prefix for Automotive and Other Applications Requiring Unique Site and Control Change Requirements; AEC-Q100 Qualified and PPAP Capable
- These Devices are Pb-Free, Halogen Free/BFR Free and are RoHS Compliant



ON Semiconductor®

<http://onsemi.com>



PDIP-8  
N, AN, VN SUFFIX  
CASE 626

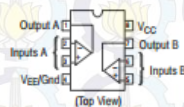


SOIC-8  
D, VD SUFFIX  
CASE 751



Micro8™  
DMR2 SUFFIX  
CASE 846A

### PIN CONNECTIONS



### ORDERING INFORMATION

See detailed ordering and shipping information in the package dimensions section on page 10 of this data sheet.

### DEVICE MARKING INFORMATION

See general marking information in the device marking section on page 11 of this data sheet.



# LM258, LM358, LM358A, LM2904, LM2904A, LM2904V, NCV2904, NCV2904V

## ELECTRICAL CHARACTERISTICS (V<sub>CC</sub> = 5.0 V, V<sub>EE</sub> = GND, T<sub>A</sub> = 25°C, unless otherwise noted.)

Characteristic	Symbol	LM258			LM358			LM358A			Unit
		Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
Input Offset Voltage V <sub>CC</sub> = 5.0 V to 30 V, V <sub>ID</sub> = 0 V to V <sub>CC</sub> = 1.7 V, V <sub>O</sub> = 1.4 V, R <sub>S</sub> = 0 Ω T <sub>A</sub> = 25°C T <sub>A</sub> = T <sub>High</sub> (Note 5) T <sub>A</sub> = T <sub>Low</sub> (Note 5)	V <sub>ID</sub>	-	2.0	5.0	-	2.0	7.0	-	2.0	3.0	mV
Average Temperature Coefficient of Input Offset Voltage T <sub>A</sub> = T <sub>High</sub> to T <sub>Low</sub> (Note 5)	ΔV <sub>ID</sub> /ΔT	-	7.0	-	-	7.0	-	-	7.0	-	μV/°C
Input Offset Current T <sub>A</sub> = T <sub>High</sub> to T <sub>Low</sub> (Note 5)	I <sub>IO</sub>	-	3.0	30	-	5.0	50	-	5.0	30	nA
Input Bias Current T <sub>A</sub> = T <sub>High</sub> to T <sub>Low</sub> (Note 5)	I <sub>IB</sub>	-	-45	-150	-	-45	-250	-	-45	-100	nA
Average Temperature Coefficient of Input Offset Current T <sub>A</sub> = T <sub>High</sub> to T <sub>Low</sub> (Note 5)	ΔI <sub>IO</sub> /ΔT	-	10	-	-	10	-	-	10	-	pA/°C
Input Common Mode Voltage Range (Note 6), V <sub>CC</sub> = 30 V V <sub>CC</sub> = 30 V, T <sub>A</sub> = T <sub>High</sub> to T <sub>Low</sub>	V <sub>ICR</sub>	0	-	28.3	0	-	28.3	0	-	28.5	V
Differential Input Voltage Range	V <sub>IDR</sub>	-	-	V <sub>CC</sub>	-	-	V <sub>CC</sub>	-	-	V <sub>CC</sub>	V
Large Signal Open Loop Voltage Gain R <sub>L</sub> = 2.0 kΩ, V <sub>CC</sub> = 15 V, For Large V <sub>O</sub> Swing, T <sub>A</sub> = T <sub>High</sub> to T <sub>Low</sub> (Note 5)	A <sub>VOL</sub>	50	100	-	25	100	-	25	100	-	V/mV
Channel Separation 1.0 kHz ≤ f ≤ 20 kHz, Input Referenced	CS	-	-120	-	-	-120	-	-	-120	-	dB
Common Mode Rejection R <sub>S</sub> ≤ 10 kΩ	CMR	70	85	-	65	70	-	65	70	-	dB
Power Supply Rejection	PSR	65	100	-	65	100	-	65	100	-	dB
Output Voltage-High Limit T <sub>A</sub> = T <sub>High</sub> to T <sub>Low</sub> (Note 5) V <sub>CC</sub> = 5.0 V, R <sub>L</sub> = 2.0 kΩ, T <sub>A</sub> = 25°C V <sub>CC</sub> = 30 V, R <sub>L</sub> = 2.0 kΩ V <sub>CC</sub> = 30 V, R <sub>L</sub> = 10 kΩ	V <sub>OH</sub>	3.3	3.5	-	3.3	3.5	-	3.3	3.5	-	V
Output Voltage-Low Limit V <sub>CC</sub> = 5.0 V, R <sub>L</sub> = 10 kΩ, T <sub>A</sub> = T <sub>High</sub> to T <sub>Low</sub> (Note 5)	V <sub>OL</sub>	-	5.0	20	-	5.0	20	-	5.0	20	mV
Output Source Current V <sub>ID</sub> = +1.0 V, V <sub>CC</sub> = 15 V T <sub>A</sub> = T <sub>High</sub> to T <sub>Low</sub> (LM358A Only)	I <sub>O+</sub>	20	40	-	20	40	-	20	40	-	mA
Output Sink Current V <sub>ID</sub> = -1.0 V, V <sub>CC</sub> = 15 V T <sub>A</sub> = T <sub>High</sub> to T <sub>Low</sub> (LM358A Only) V <sub>ID</sub> = -1.0 V, V <sub>O</sub> = 200 mV	I <sub>O-</sub>	10	20	-	10	20	-	10	20	-	mA
Output Short Circuit to Ground (Note 7)	I <sub>SC</sub>	-	40	60	-	40	60	-	40	60	mA
Power Supply Current (Total Device) T <sub>A</sub> = T <sub>High</sub> to T <sub>Low</sub> (Note 5) V <sub>CC</sub> = 30 V, V <sub>O</sub> = 0 V, R <sub>L</sub> = ∞ V <sub>CC</sub> = 5 V, V <sub>O</sub> = 0 V, R <sub>L</sub> = ∞	I <sub>CC</sub>	-	1.5	3.0	-	1.5	3.0	-	1.5	2.0	mA

5. LM258: T<sub>Low</sub> = -25°C, T<sub>High</sub> = +85°C  
LM2904/LM2904A: T<sub>Low</sub> = -40°C, T<sub>High</sub> = +105°C  
LM358, LM358A: T<sub>Low</sub> = 0°C, T<sub>High</sub> = +70°C  
LM2904V & NCV2904: T<sub>Low</sub> = -40°C, T<sub>High</sub> = +125°C  
NCV2904V & NCV2904V are qualified for automotive use. NCV2904V: T<sub>Low</sub> = -40°C, T<sub>High</sub> = +150°C
6. The input common mode voltage or either input signal voltage should not be allowed to go negative by more than 0.3 V. The upper end of the common mode voltage range is V<sub>CC</sub> = 1.7 V.
7. Short circuits from the output to V<sub>CC</sub> can cause excessive heating and eventual destruction. Destructive dissipation can result from simultaneous shorts on all amplifiers.



## RIWAYAT HIDUP



Nama : Kresna Tri Jayanto  
Siswadi  
Tempat Lahir : Bandung  
Tanggal Lahir : 9 Januari 1992  
Agama : Islam  
Alamat Rumah : Desa Tlogo 2 Rt2 Rw4  
No.8 Kec Kanigoro  
Blitar, Bojonegoro  
Telp/HP : 085330886762  
E-mail : Kresnatri@Gmail.Com

### RIWAYAT PENDIDIKAN

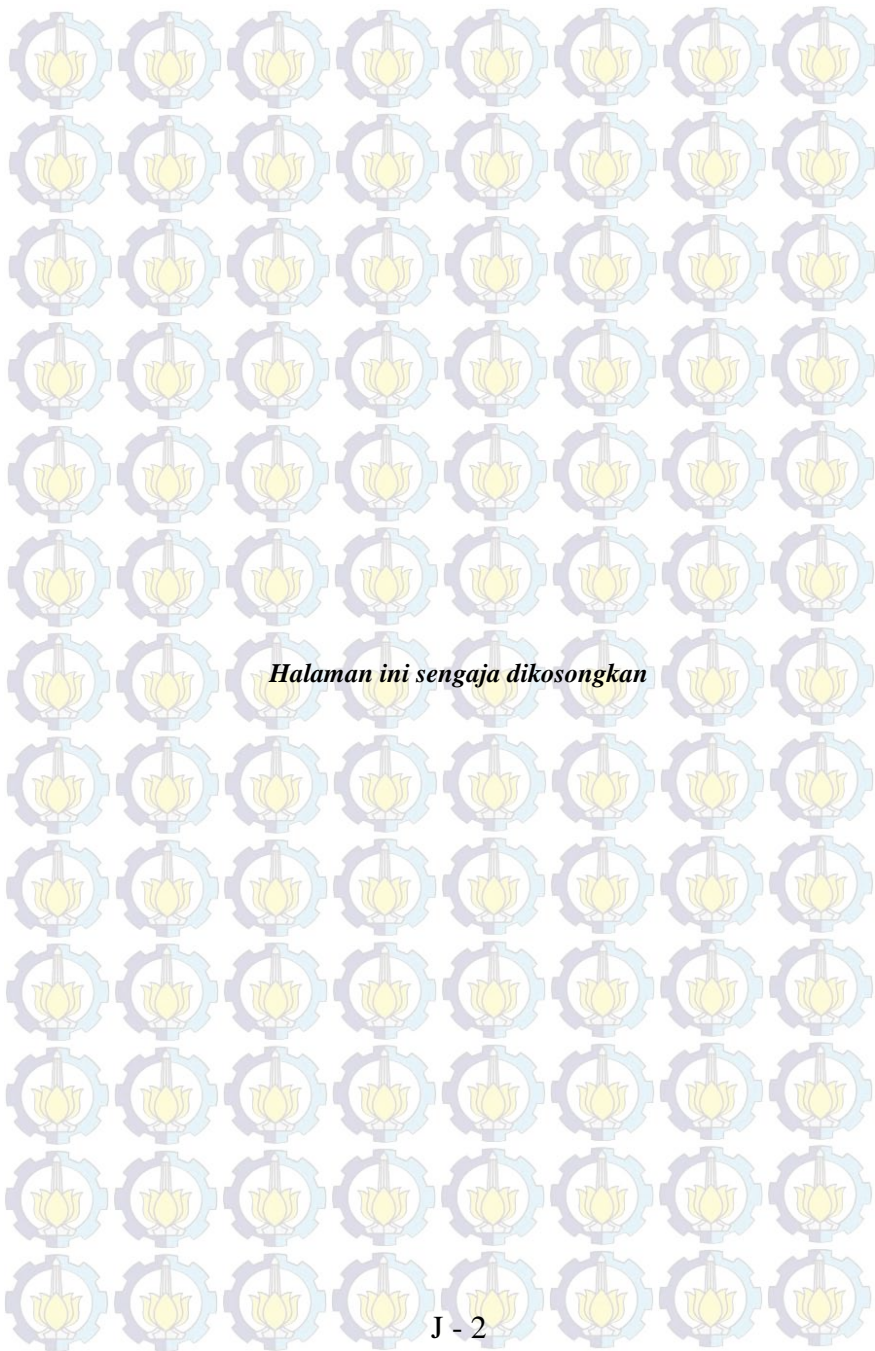
TK Dharma Wanita (1997-1999)  
SDN Tlogo 1 (1999-2005)  
SMPN 3 Blitar (2005-2008)  
SMAN 1 Blitar (2008-2011)  
ITS (2011-Sekarang)

### PENGALAMAN KERJA

Kerja Praktek di PT.TELKOMSEL Gayungan - Surabaya.

### PENGALAMAN ORGANISASI

Seksi materi pada UKM fotografi SMA Negeri 1 Blitar



## RIWAYAT HIDUP



Nama : Muhammad Ihya  
Uddin Al - Karimi  
Tempat Lahir : Surabaya  
Tanggal Lahir : 07 Agustus 1993  
Agama : Islam  
Alamat Rumah : Jl.Sikatan  
15/13,Tandes,Surabaya  
Telp/HP : 083830200892  
E-mail : Elkarimi@Ymail.Com

### RIWAYAT PENDIDIKAN

TK Ta'Miriyah (1997-1999)  
SD Ta'Miriyah (1999-2005)  
SMPM 12 (2005-2008)  
SMA Hidayatullah (2008-2011)  
ITS (2011-Sekarang)

### PENGALAMAN KERJA

Kerja Praktek di PT.PAL Jl.Ujung - Surabaya.

### PENGALAMAN ORGANISASI

Anggota Kominfo Himad3teknikelektro  
Staff Lemabga Dakwah Jurusan D3teknikelektro



